Voice API Implementation Examples

This document provides concrete implementation examples for the Voice API integration. These examples demonstrate how to use the Voice API in different contexts and scenarios.

Table of Contents

Backend Examples

Voice Synthesis

This example demonstrates how to use the Voice Controller to synthesize speech from text.

```
 // src/backend/controllers/voiceController.js

/**
 * Synthesize text into speech
 *
 * @param {Object} req - Express request object
 * @param {Object} res - Express response object
 */
exports.synthesizeSpeech = async (req, res) => {
  try {
    const { text, voiceId, outputFormat } = req.body;

    if (!text) {
      return res.status(400).json({ error: 'Text is required' });
    }

    logger.info(`Synthesizing speech: "${text.substring(0, 50)}...

    // Call the AI service
    const response = await axios.post(
      `${config.aiServiceUrl}/voice/synthesize`,
      {
        text,
        voice_id: voiceId,
        output_format: outputFormat || 'mp3'
      },
      {
        headers: {
          'Content-Type': 'application/json',
          'x-api-key': config.aiServiceApiKey
        }
      }
    );

    // Return the response from the AI service
    return res.status(200).json(response.data);
  } catch (error) {
    logger.error(`Error in synthesizeSpeech: ${error.message}`);
    return res.status(500).json({ error: 'Internal server error' 
  }
};
```

Voice Recognition

This example demonstrates how to use the Voice Controller to recognize speech from audio data.

```
// src/backend/controllers/voiceController.js

/**
 * Recognize speech from audio data
 *
 * @param {Object} req - Express request object
 * @param {Object} res - Express response object
 */
exports.recognizeSpeech = async (req, res) => {
  try {
    const { audioData, language } = req.body;

    if (!audioData) {
      return res.status(400).json({ error: 'Audio data is required
    }

    logger.info(`Recognizing speech with language: ${language ||

    // Call the AI service
    const response = await axios.post(
      `${config.aiServiceUrl}/voice/recognize`,
      {
        audio_data: audioData,
        language: language || 'en-US'
      },
      {
        headers: {
          'Content-Type': 'application/json',
          'x-api-key': config.aiServiceApiKey
        }
      }
    );

    // Return the response from the AI service
    return res.status(200).json(response.data);
  } catch (error) {
    logger.error(`Error in recognizeSpeech: ${error.message}`);
    return res.status(500).json({ error: 'Internal server error' 
  }
};
```

Frontend Examples

Recording Voice

This example demonstrates how to use the VoiceRecorder component to record voice input.

```
// Example component using VoiceRecorder
import React, { useState } from 'react';
import VoiceRecorder from '../components/VoiceRecorder';

const VoiceInputExample = () => {
  const [recognizedText, setRecognizedText] = useState('');

  const handleAudioCaptured = (text, audioData) => {
    setRecognizedText(text);
    // You can also send the audio data to the server for further
  };

  return (
    <div className="voice-input-example">
      <h2>Voice Input Example</h2>

      <VoiceRecorder onAudioCaptured={handleAudioCaptured} />

      {recognizedText && (
        <div className="recognized-text">
          <h3>Recognized Text:</h3>
          <p>{recognizedText}</p>
        </div>
      )}
    </div>
  );
};

export default VoiceInputExample;
```

Playing Synthesized Speech

This example demonstrates how to use the VoicePlayback component to play synthesized speech.

```
// Example component using VoicePlayback
import React, { useState } from 'react';
import VoicePlayback from '../components/VoicePlayback';

const VoiceOutputExample = () => {
  const [text, setText] = useState('');
  const [audioData, setAudioData] = useState(null);

  const handleTextChange = (e) => {
    setText(e.target.value);
  };

  const handleGenerateSpeech = async () => {
    if (!text) return;

    try {
      const response = await fetch('/api/voice/synthesize', {
        method: 'POST',
        headers: {
          'Content-Type': 'application/json',
          'Authorization': `Bearer ${localStorage.getItem('token'
        },
        body: JSON.stringify({
          text,
          voiceId: 'en-US-Standard-A',
          outputFormat: 'mp3'
        })
      });

      const data = await response.json();

      if (data.audio_data) {
        setAudioData(data.audio_data);
      }
    } catch (error) {
      console.error('Error generating speech:', error);
    }
  };
  return (
    <div className="voice-output-example">
      <h2>Voice Output Example</h2>

      <div className="text-input">
        <textarea
          value={text}
          onChange={handleTextChange}
          placeholder="Enter text to synthesize"
          rows={4}
        />

        <button
          onClick={handleGenerateSpeech}
          disabled={!text}
        >
          Generate Speech
        </button>
      </div>

      {audioData && (
        <div className="playback-container">
          <h3>Generated Speech:</h3>

          <VoicePlayback
            audioData={audioData}
            autoPlay={true}
          />
        </div>
      )}
    </div>
  );
};

export default VoiceOutputExample;
```

AI Service Examples

Voice Service Integration

This example demonstrates how to integrate the Voice Service with external voice APIs.

```python
```

src/ai/voice_service.py

async def synthesize_speech(self, text: str, voice_id: Optional[str] = None, output_format: str = "mp3") -> Dict[str, Any]: """ Synthesize text into speech.

```
 Args:
    text: Text to synthesize
    voice_id: Voice ID to use
    output_format: Output audio format

Returns:
    Dictionary containing the audio data and metadata
"""
logger.info(f"Synthesizing speech: '{text[:50]}...' with voice: {

try:
    # Select the appropriate TTS service
    if self.tts_service == "google":
        audio_data, duration = await self._synthesize_google(text
    elif self.tts_service == "amazon":
        audio_data, duration = await self._synthesize_amazon(text
    elif self.tts_service == "microsoft":
        audio_data, duration = await self._synthesize_microsoft(te
    else:
        # Use mock implementation for testing
        audio_data, duration = await self._synthesize_mock(text,

    return {
        "audio_data": audio_data,
        "format": output_format,
        "duration": duration,
        "text": text
    }
except Exception as e:
    logger.error(f"Error in speech synthesis: {e}")
    raise HTTPException(status_code=500, detail=f"Speech synthesis
```

Generated 2026-01-13 06:26 UTC