TTSmate V1 测试文档

版本: 1.1.0

测试目的:

验证 TTSmate 应用程序的核心功能、UI 交互、API 集成和错误处理符合设计要求。

测试环境:

操作系统: Windows

依赖: 互联网连接

测试前准备:

1. 确保已编译生成最新的 TTSmateV1.exe 文件。

2. 准备有效的 DeepSeek API Key 以及百度的 API Key 和 Secret Key。

3. 在 sounds/ 目录下放置至少一个有效的音效文件 (例如 sound1.wav)。

4. 准备一个无效或损坏的音效文件用于错误测试。

一、 启动与配置测试

测试用例 1.1: 配置文件缺失

目标: 验证应用在 config.toml 文件不存在时的行为。

步骤:

1. 确保工作目录下没有 config.toml 文件。

2. 启动应用程序。

预期结果:

应用程序启动失败，并在控制台或日志中打印出明确的错误信息，提示 "Failed to load config.toml"。

实际结果:

状态 (通过/失败):

测试用例 1.2: 配置文件正常加载

目标: 验证应用能成功加载有效的配置文件并启动。

步骤:

1. 创建一个有效的 config.toml 文件，并填入所有必需的密钥和设置。

2. 启动应用程序。

预期结果:

应用程序正常启动，显示主界面。

实际结果:

状态 (通过/失败):

测试用例 1.3: 配置文件部分字段缺失

目标: 验证应用对不完整配置的容错能力。

步骤:

1. 编辑 config.toml，删除 [ai\_settings] 或 [[soundboard]] 等非关键部分。

2. 启动应用程序。

预期结果:

应用程序应该能正常启动。对于缺失的部分，相关功能（如AI模板、音效板）将为空或使用默认值，不应导致程序崩溃。

实际结果:

状态 (通过/失败):

二、 UI 与核心功能测试

测试用例 2.1: 直接文本转语音

目标: 验证禁用 AI 时，应用能直接将输入文本进行语音合成。

步骤:

1. 取消勾选 "使用 DeepSeek 生成文案" 复选框。

2. 在 "输入话题/文本" 框中输入 "你好世界"。

3. 点击 "生成并播放" 按钮。

预期结果:

状态栏显示 "正在合成语音..." -> "正在播放..." -> "就绪"。

可以听到 "你好世界" 的语音播报。

"AI生成文本" 区域不更新。

实际结果:

状态 (通过/失败):

测试用例 2.2: AI 文本生成与播放

目标: 验证启用 AI 时，应用能根据主题生成文本并播放。

步骤:

1. 勾选 "使用 DeepSeek 生成文案" 复选框。

2. 在 "输入话题/文本" 框中输入一个主题，例如 "宇宙"。

3. 点击 "生成并播放" 按钮。

预期结果:

状态栏显示 "正在生成文本..." -> "正在合成语音..." -> "正在播放..." -> "就绪"。

"AI生成文本" 区域显示一段与 "宇宙" 相关的内容。

可以听到 AI 生成文本的语音播报。

实际结果:

状态 (通过/失败):

测试用例 2.3: AI 提示词模板切换

目标: 验证切换 AI 模板能改变 AI 的行为。

步骤:

1. 勾选 "使用 DeepSeek 生成文案"。

2. 在 "AI 设置" 中，选择 "游戏旁白" 模板。

3. 在输入框输入 "一个古老的城堡"。

4. 点击 "生成并播放"。

预期结果:

AI 生成的文本和语音带有明显的神秘或游戏旁白风格。

实际结果:

状态 (通过/失败):

测试用例 2.4: 语音参数调整

目标: 验证语速、音调、音量、发音人参数的调整能实时生效。

步骤:

1. 输入任意文本并点击 "生成并播放"，听取默认效果。

2. 将语速、音调、音量滑块调整到不同位置。

3. 切换一个不同的发音人。

4. 再次点击 "生成并播放"。

预期结果:

后续播放的语音在语速、音调、音量和音色上与第一次有明显不同，符合调整预期。

实际结果:

状态 (通过/失败):

三、 音效板功能测试

测试用例 3.1: 播放有效音效

目标: 验证音效板能成功加载并播放有效的音效文件。

步骤:

1. 在 config.toml 中配置一个指向有效音频文件 (如 sound1.wav) 的音效。

2. 启动应用。

3. 在 "音效板" 区域点击对应的按钮。

预期结果:

可以立即听到对应的音效。

主状态栏的文本不应改变。

实际结果:

状态 (通过/失败):

测试用例 3.2: 音效与 TTS 并行播放

目标: 验证音效可以在 TTS 语音播放时同时播放。

步骤:

1. 点击 "生成并播放" 播放一段较长的 TTS 语音。

2. 在 TTS 语音仍在播放时，点击音效板上的一个音效按钮。

预期结果:

可以同时听到 TTS 语音和音效的声音。

实际结果:

状态 (通过/失败):

测试用例 3.3: 播放缺失的音效文件

目标: 验证当音效文件不存在时，应用不会崩溃。

步骤:

1. 在 config.toml 中配置一个指向不存在路径的音效。

2. 启动应用并点击该音效按钮。

预期结果:

应用无响应，没有声音播放，不会崩溃。

控制台或日志中应出现 "读取音效文件 ... 失败" 的错误信息。

实际结果:

状态 (通过/失败):

四、 错误处理与边界测试

测试用例 4.1: 无效的 API 密钥

目标: 验证当 API 密钥无效时，应用能正确处理错误。

步骤:

1. 在 config.toml 中填入一个无效的 DeepSeek 或 Baidu API 密钥。

2. 启动应用，尝试使用依赖该密钥的功能。

预期结果:

状态栏显示对应的错误信息，例如 "错误: DeepSeek API 调用失败: ..." 或 "错误: Baidu TTS API 调用失败: ..."。

应用不会崩溃。

实际结果:

状态 (通过/失败):

测试用例 4.2: 输入为空

目标: 验证对空输入文本的处理。

步骤:

1. 保持输入框为空。

2. 点击 "生成并播放"。

预期结果:

状态栏显示错误信息 "错误: 无有效文本可供合成"。

没有 API 请求发出。

实际结果:

状态 (通过/失败):

测试用例 4.3: 无网络连接

目标: 验证在断网情况下的行为。

步骤:

1. 断开电脑的网络连接。

2. 启动应用，点击 "生成并播放"。

预期结果:

状态栏在短暂延迟后显示 API 调用失败的错误信息。

应用不会崩溃。

实际结果:

状态 (通过/失败):