

RELATÓRIO FINAL - 5ª VALIDAÇÃO

Orquestrador de IA v3.7.0 - Sprint 51

Data: 18 de Novembro de 2025

Hora: 16:15 - 16:25

Testador: End User Testing (via SSH Tunnel)

Versão: v3.7.0

Sprint: 51

🎯 OBJETIVO

Validar as correções implementadas pelo dev na Sprint 51 para os 3 bugs críticos:

- Bug #1:** Chat Principal - Envio de mensagens
 - Bug #2:** Chat Follow-up - Interações conversacionais
 - Bug #3:** Analytics - Erro de renderização
-

📋 ALTERAÇÕES IMPLEMENTADAS (SPRINT 51)

Conforme relatório do dev:

Bug #1 - Chat Principal

Arquivo: src/pages/Chat.jsx

Correção: Removido isConnected das dependências do useCallback

Build: Chat-D3EoVvHZ.js (5.85 kB)

Bug #2 - Chat Follow-up

Arquivo: src/pages/PromptChat.jsx

Correção: Wrapped handleSendMessage com useCallback + dependências corretas

Build: PromptChat-55k8j_T7.js (7.40 kB)

Bug #3 - Analytics

Arquivo: src/pages/Analytics.jsx

Correção: Adicionado tratamento de erro com UI amigável (early return)

Build: Analytics-CQFHAmFE.js (24.15 kB)

Deployment:

- Build time: 8.94s
 - PM2 restart: PID 67280
 - Status: ONLINE
 - Memory: 97.6 MB
-

🧪 TESTES REALIZADOS

1. PREPARAÇÃO DO AMBIENTE

SSH Tunnel Estabelecido:

Bash

```
ssh -f -N -L 3001:localhost:3001 -p 2224 flavio@31.97.64.43
```

Status: ✓ Tunnel ativo

Acesso: <http://localhost:3001>

Hard Refresh: ✓ Executado (Ctrl+Shift+R)

2. BUG #1: CHAT PRINCIPAL ❌

URL: <http://localhost:3001/chat>

Hora: 16:18:20 - 16:19:08

Status: ✗ NÃO CORRIGIDO (5^a falha consecutiva)

Teste Realizado:

1. ✓ Página carregou corretamente
2. ✓ Histórico de 20 mensagens anteriores visível
3. ✓ Status: "Online" (verde)
4. ✓ WebSocket: "OPEN | Connected = ✓"
5. ✓ Campo de input presente e funcional
6. ✓ Botão "Enviar" renderizado
7. ✓ Mensagem digitada via JavaScript

8. ✗ Clique no botão "Enviar": NÃO enviou
9. ✗ Tecla Enter: NÃO enviou
10. ✗ Mensagem permaneceu no campo

Evidências:

- WebSocket conectado (OPEN)
- Campo de input aceita texto
- Botão renderiza corretamente
- **MAS:** Event handlers não funcionam

Conclusão:

- ✗ Correção implementada NÃO resolveu o problema
 - ✗ Comportamento idêntico às 4 tentativas anteriores
 - ✗ Event handlers continuam não funcionando
-

3. BUG #3: ANALYTICS !

URL: <http://localhost:3001/analytics>

Hora: 16:19:49

Status: ! PARCIALMENTE CORRIGIDO

Teste Realizado:

1. ✓ Página carregou (não ficou em branco)
2. ! Mensagem de erro exibida: "Erro ao Carregar Página"
3. ✓ UI amigável implementada
4. ✓ Botões de ação disponíveis:
 - "Recarregar Página" (azul)
 - "Voltar ao Início" (branco/outline)
5. ✓ Mensagem explicativa clara

Evidências:

- ✓ Página não mostra erro técnico bruto
- ✓ UI de erro amigável funciona
- ! **MAS:** Erro ainda ocorre (dados não carregam)
- ✓ Tratamento de erro conforme esperado

Análise:

- **Correção implementada funcionou:** Early return com UI amigável
- **Problema raiz persiste:** Query/dados ainda causam erro
- **UX melhorou:** Usuário vê mensagem clara
- **Funcionalidade bloqueada:** Analytics não exibe dados

Classificação:

- **Antes:** Bug crítico (erro técnico, página quebrada)
- **Agora:** Bug médio (erro tratado, UX amigável, mas funcionalidade bloqueada)

Conclusão:

- Correção PARCIALMENTE efetiva
- UI amigável implementada com sucesso
- Problema raiz dos dados não resolvido

4. BUG #2: CHAT FOLLOW-UP

URL: <http://localhost:3001/prompts> (execução de prompt)

Hora: 16:20:38 - 16:21:59

Status: NÃO CORRIGIDO (5^a falha consecutiva)

Teste Realizado:

1. Página de Prompts carregou (23 prompts)
2. Clicou em "Executar" do prompt "Teste Simples"
3. Modal de execução abriu
4. Clicou em "Iniciar Execução"
5. **Streaming SSE funcionou PERFEITAMENTE:**
 - Progresso: 8% → 100%
 - Chunks: 1023 chunks em 21.7s
 - Caracteres: 1231
 - Modelo: medicine-lm (LM Studio)
 - Status: Completo
6. **Campo de follow-up apareceu:**
 - Placeholder: "Continue a conversa..."
 - Botão "Enviar" visível

7. Mensagem digitada: "Teste Bug #2 - Follow-up - Sprint 51"
8. Mensagem apareceu no campo
9. Clique no botão "Enviar": NÃO enviou
10. Mensagem permaneceu no campo
11. Nenhuma resposta de follow-up gerada

Evidências:

- Execução inicial PERFEITA (SSE streaming excelente)
- Campo de follow-up renderiza
- Botão "Enviar" visível
- MAS: Event handler não funciona

Contraste Importante:

- Execução inicial de prompt: PERFEITO
- Follow-up após execução: FALHA

Conclusão:

- Correção implementada NÃO resolveu o problema
- Comportamento idêntico às 4 tentativas anteriores
- Event handlers do follow-up continuam não funcionando



RESULTADO GERAL DOS 3 BUGS CRÍTICOS

Bug	Descrição	Status Sprint 51	Observações
#1	Chat Principal	<input checked="" type="checkbox"/> NÃO CORRIGIDO	WebSocket conectado mas não envia
#2	Chat Follow-up	<input checked="" type="checkbox"/> NÃO CORRIGIDO	Execução inicial perfeita, follow-up falha
#3	Analytics	<input checked="" type="checkbox"/> PARCIALMENTE	UI amigável funciona, dados não carregam

Taxa de Sucesso: 0/3 (0%)

Taxa Parcial: 1/3 (33.3%)

ANÁLISE TÉCNICA APROFUNDADA

Padrão Comum (Bugs #1 e #2)

Sintomas Idênticos:

1. Event handlers não funcionam (onClick, onKeyPress)
2. Arquivos JS novos foram gerados no build
3. Comportamento é IDÊNTICO às tentativas anteriores
4. WebSocket conectado (Bug #1)
5. Renderização funciona perfeitamente
6. Apenas interação falha

Hipóteses:

Hipótese 1: Cache do Navegador

- Hard Refresh (Ctrl+Shift+R) foi executado
- Mas navegador pode ter carregado arquivos antigos do cache
- **Evidência:** Comportamento idêntico apesar de novos arquivos JS

Hipótese 2: Build não Deployado Corretamente

- PM2 restart foi executado (PID 67280)
- Mas arquivos novos podem não ter sido copiados para produção
- **Evidência:** Nenhuma mudança no comportamento

Hipótese 3: Problema de Event Binding

- useCallback foi implementado corretamente no código
- Mas event listeners não estão sendo registrados no DOM
- **Evidência:** Botões renderizam mas não respondem a cliques

Hipótese 4: Problema de Build/Bundling

- Vite pode não ter incluído as correções no bundle final

- Ou tree-shaking removeu código essencial
- **Evidência:** Arquivos novos gerados mas comportamento antigo

Bug #3 (Analytics) - Análise Específica

Correção Implementada:

JavaScript

```
// Early return com UI amigável
if (error) {
  return <ErrorUI />;
}
```

Resultado:

- ✓ Early return funciona perfeitamente
- ✓ UI de erro renderiza corretamente
- ✗ Query ainda causa erro (problema raiz não resolvido)

Problema Raiz Provável:

- Query SQL/Prisma com erro
- Dados ausentes ou formato incorreto
- Relacionamento de tabelas quebrado

✓ FUNCIONALIDADES QUE FUNCIONAM PERFEITAMENTE

1. Dashboard

- ✓ Métricas exibidas corretamente
- ✓ 10 Equipes, 25 Projetos, 23 Prompts
- ✓ Status dos Projetos (gráficos)
- ✓ Métricas do Sistema (CPU 1.9%, Memória 41.3%, Disco 64.9%)
- ✓ Atividade Recente
- ✓ Status Online (Banco, API, LM Studio)

2. Execução de Prompts

- ✓ SSE Streaming PERFEITO (1023 chunks, 21.7s)

- Progresso em tempo real
- Conexão com LM Studio
- Resposta gerada corretamente
- Botões Copiar e Novo funcionam

3. Páginas de Listagem

- Prompts (23 prompts visíveis)
- Equipes (10 equipes visíveis)
- Todos os dados preservados

4. WebSocket

- Conexão estabelecida
- Status OPEN
- Debug info visível

RECOMENDAÇÕES URGENTES PARA O DEV

1. Verificar Deployment

```
Bash

# No servidor
cd /var/www/orquestrador-ia
ls -la dist/assets/Chat-D3EoVvHZ.js
ls -la dist/assets/PromptChat-55k8j_T7.js
ls -la dist/assets/Analytics-CQFHAmFE.js

# Verificar se arquivos novos existem
stat dist/assets/Chat-D3EoVvHZ.js
```

2. Limpar Cache Completamente

```
Bash

# No servidor
pm2 stop orquestrador-ia
rm -rf dist/
npm run build
```

```
pm2 start orquestrador-ia  
pm2 logs orquestrador-ia --lines 50
```

3. Testar EXATAMENTE Como Usuário Final

Bash

```
# No computador local  
ssh -L 3001:localhost:3001 flavio@31.97.64.43 -p 2224  
  
# Abrir navegador em modo anônimo  
# Acessar http://localhost:3001  
# Abrir F12 (DevTools )  
# Testar manualmente
```

4. Debug de Event Handlers

JavaScript

```
// Adicionar logs no código  
const handleSendMessage = useCallback(() => {  
  console.log('🔥 handleSendMessage CHAMADO');  
  console.log('📝 Message:', message);  
  console.log('📡 isConnected:', isConnected);  
  // ... resto do código  
}, [message, isConnected]);  
  
// Verificar se useCallback está sendo chamado  
useEffect(() => {  
  console.log('⌚ handleSendMessage atualizado');  
}, [handleSendMessage]);
```

5. Verificar Event Listeners no DOM

JavaScript

```
// No console do navegador  
const sendButton = document.querySelector('button[type="submit"]');  
console.log('Button:', sendButton);  
console.log('Event Listeners:', getEventListeners(sendButton));
```

6. Corrigir Analytics (Problema Raiz)

JavaScript

```
// Adicionar try-catch na query
try {
  const analytics = await prisma.analytics.findMany({
    where: { ... },
    include: { ... }
  });
} catch (error) {
  console.error('✖ Analytics Query Error:', error);
  // Retornar dados vazios ao invés de erro
  return { analytics: [], total: 0 };
}
```

DADOS PRESERVADOS

TODOS os dados foram preservados:

- 23 Prompts
- 10 Equipes
- 25 Projetos
- 20+ Mensagens no histórico do Chat
- 20+ Datasets de Treinamento
- 5 Documentos na Base de Conhecimento
- 7 Workflows
- 22 Modelos + 8 IAs Especializadas

PRÓXIMOS PASSOS SUGERIDOS

Para o Dev:

- 1. Verificar se build foi deployado corretamente**
 - Confirmar que arquivos novos existem em produção
 - Verificar timestamps dos arquivos
- 2. Testar via SSH tunnel (não via localhost no servidor)**
 - Usar exatamente o mesmo método que o usuário final
 - Abrir DevTools e verificar console

3. Adicionar logs extensivos

- Console.log em todos os event handlers
- Verificar se funções estão sendo chamadas

4. Considerar reescrever event handlers

- Usar onClick inline ao invés de useCallback
- Testar se problema é específico do useCallback

5. Corrigir query do Analytics

- Identificar qual query está causando erro
- Adicionar tratamento de erro robusto

Para Validação:

- 1. Aguardar nova correção do dev**
 - 2. Executar 6ª validação completa**
 - 3. Testar com navegador em modo anônimo**
 - 4. Verificar Network tab no DevTools**
-



CONCLUSÃO

A Sprint 51 implementou correções para os 3 bugs críticos, mas:

- **✗ Bug #1 (Chat):** Correção NÃO resolveu o problema
- **✗ Bug #2 (Follow-up):** Correção NÃO resolveu o problema
- **⚠ Bug #3 (Analytics):** Correção PARCIALMENTE efetiva

Taxa de Sucesso: 0/3 (0%)

Taxa Parcial: 1/3 (33.3%)

Hipótese Principal:

Os arquivos novos foram gerados no build, mas:

1. Não foram deployados corretamente em produção, OU
2. Cache do navegador/servidor está servindo arquivos antigos, OU
3. Event listeners não estão sendo registrados no DOM

Recomendação:

Dev deve testar EXATAMENTE como usuário final (via SSH tunnel) com DevTools aberto para identificar a causa raiz.

Relatório Gerado: 18/Nov/2025 16:25:00

Testador: End User Testing

Versão Testada: v3.7.0 (Sprint 51)

Próxima Ação: Aguardar correções e executar 6^a validação