

Guia de Testes do Sistema de Gestão de Compras (SLA)

Este guia orienta testes ponta a ponta por página e por perfil de usuário, garantindo cobertura funcional, acesso por perfil e persistência de sessão. Referências ao código entre crases apontam para trechos em `app.py`.

Visão Geral de Perfis e Navegação

- Em `app.py`, a navegação por perfil é definida em `opcoes_por_perfil()`:
- Solicitante: `■ Nova Solicitação`, `■ Dashboard SLA`, `■ Histórico por Etapa`
- Suprimentos: `■ Mover para Próxima Etapa`, `■ Dashboard SLA`, `■ Histórico por Etapa`
- Gerência&Diretoria: `■ Aprovações`, `■ Dashboard SLA`, `■ Histórico por Etapa`
- Admin: todas as opções, incluindo `■ Gerenciar Usuários`

Caminho: `app.py` > função `opcoes_por_perfil()` (linhas ~657-687).

- Logout: botão `■ Logout` no sidebar. Caminho: `app.py` (linhas ~632-643).
- Persistência de sessão (Cloud): cookie assinado via `simple_session.py` e `ensure_session_persistence()`. O segredo `cookie_secret` deve estar configurado no `secrets.toml` do Streamlit Cloud.

Preparação

- Faça login como Admin padrão: usuário `admin`, senha `admin123`. O admin é criado automaticamente no primeiro run.

Caminho: `app.py` > `ensure_admin_user()` (linhas ~241-255).

- Com Admin, crie usuários de teste para cada perfil em `■ Gerenciar Usuários`.

Sugestão de usuários:

- solicitante1 / senha: 123456, Perfil: Solicitante
- suprimentos1 / senha: 123456, Perfil: Suprimentos
- gestao1 / senha: 123456, Perfil: Gerência&Diretoria;

Testes de Autenticação e Sessão

- Login:
- Acesse o formulário no sidebar: campos "Usuário" e "Senha".

Caminho: `app.py` (linhas ~596-606).

- Esperado: após login, aparece o bloco "■ Usuário" com Nome e Perfil.
- Persistência (Cloud):
- Atualize a página (F5) após login.
- Esperado: permanece logado (sessão restaurada do cookie).
- Logout:
- Clique `■ Logout`.
- Esperado: sessão encerrada; requer login novamente.

Cenário E2E (fluxo completo)

1) Como Solicitante (solicitante1):

- Vá em **■ Nova Solicitação**.

Campos:

- Solicitante (Nome e Sobrenome)*
- Departamento* (lista `DEPARTAMENTOS` em `app.py` linhas 50-58)
- Prioridade* (lista `PRIORIDADES` em `app.py` linhas 60-65)
- Descrição*
- Aplicação (Código)*
- Anexos (opcional, tipos permitidos em `ALLOWED_FILE_TYPES` — `app.py` linhas 75-77)
- Informações de controle (somente leitura): Nº Solicitação, Status Inicial, Data/Hora, SLA

O SLA exibido segue `SLA_PADRAO` (`app.py` linhas 67-73) e `obter_sla_por_prioridade()` (`app.py` linhas 316-319).

- Clique **■ Criar Solicitação**.
- Esperado:
- Mensagem de sucesso com número gerado, status “Solicitação”, SLA aplicado, qtd. anexos.
- Registro salvo em `data["solicitacoes"]` com histórico inicial.

Código de criação: `app.py` (linhas ~851-931 e 865-901).

2) Como Suprimentos (suprimentos1):

- Vá em **■ Mover para Próxima Etapa**.
- Selecione a solicitação ativa (não “Pedido Finalizado”).
- Passo “Suprimentos”: informar “Responsável Suprimentos” e Observações (opcional). Salve.
- Passo “Em Cotação”: informe “Nº Pedido (Compras)”, “Data Nº Pedido”, “Data Cotação” e Observações. Salve.
- Passo “Aguardando Aprovação”: cadastre cotações (mín. 1; ideal 3) com Fornecedor, Valor (>0), Prazo, Validade e Anexos. Salve.
- Esperado:
- Histórico de etapas atualizado a cada avanço.
- Em “Aguardando Aprovação”, as cotações são salvas, “fornecedor_recomendado” e “valor_estimado” são preenchidos com a melhor cotação, e notificação enviada para “Gerência&Diretoria”.

Caminho: `app.py` (linhas ~1072-1107 e ~1126-1131).

- SLA/dias decorridos exibidos no topo e aviso “Dentro do SLA” ou “SLA Estourado” (`calcular_dias_uteis()` — `app.py` linhas 283-307).

3) Como Gerência&Diretoria; (gestao1):

- Vá em **■ Aprovações**.
- Se existirem solicitações em “Aguardando Aprovação”, selecione e visualize:
- Prioridade, SLA (dias), Dias decorridos, Anexos
- Descrição
- Comentários (opcional)
- Realize:
- **■ Aprovar**: status vai para “Aprovado”, histórico atualizado, notificação para “Suprimentos”.

- “■ Reprovar”: status “Reprovado”, histórico atualizado, notificação para “Solicitante”.
- Esperado:
- Mensagens: “Solicitação aprovada com sucesso! Avance para 'Pedido Finalizado'” ou “Solicitação reprovada.”

Caminho: `app.py` (linhas ~1157-1232).

4) Como Suprimentos (finalização):

- Retorne a `■ Mover para Próxima Etapa` na solicitação aprovada.
- Avance para “Pedido Finalizado” informando:
- Data Entrega*
- Valor Final (R\$)
- Fornecedor Final
- Observações Finais
- Esperado:
- Cálculo de `dias_atendimento` e `sla_cumprido` (“Sim/Não”) com base em dias úteis (`verificar_sla_cumprido()` em `app.py` linhas 309-315).
- Exibição: “■ SLA CUMPRIDO!” ou “■■ SLA NÃO CUMPRIDO!”.

Caminho: `app.py` (linhas ~1108-1152).

5) Validação nos relatórios:

- `■ Dashboard SLA`: valide métricas, distribuição por etapa, performance por departamento e prioridade, e “Solicitações com SLA em Risco”.
- `■ Histórico por Etapa`: filtre por etapa, departamento, prioridade; baixe CSV e verifique colunas esperadas: Nº, Data/Hora, Solicitante, Departamento, Prioridade, Descrição (truncada), Status, SLA, Dias Atendimento, SLA Cumprido, Nº Pedido, Data Entrega.

Caminho: `app.py` (linhas ~1462-1534).

Testes por Página

- ■ Nova Solicitação
- Campos obrigatórios: nome, departamento, prioridade, descrição, aplicação.
- SLA exibido dinamicamente segundo prioridade (`obter_sla_por_prioridade()`).
- Uploads múltiplos; após envio, a área mostra “■ X arquivo(s) selecionado(s)”.
- Esperado: número de solicitação incremento automático, histórico inicial, anexos salvos em subpasta da solicitação (quando aplicável).
- Código: `app.py` (linhas ~734-931).
- ■ Mover para Próxima Etapa
- Só lista solicitações não finalizadas (“Pedido Finalizado”).
- Mostra métricas de status, prioridade, SLA e dias decorridos.
- Próximas etapas possíveis:
- Solicitação → Suprimentos
- Suprimentos → Em Cotação
- Em Cotação → Aguardando Aprovação
- Aguardando Aprovação → (apenas via página `■ Aprovações`)

- Aprovado → Pedido Finalizado
- Reprovado → sem próxima etapa (a tela pode indicar que está finalizada)
- Esperado: histórico e campos específicos de cada etapa atualizados.
- Código: `app.py` (linhas ~932-1156).
- ■ Aprovações
 - Acesso restrito: apenas “Gerência&Diretoria;” e “Admin” (o sidebar não exibe esta opção para outros perfis).
 - Aprovação muda status para “Aprovado”; reprovação para “Reprovado”; histórico e notificações atualizados.
 - Código: `app.py` (linhas ~1157-1232).
- ■ Dashboard SLA
 - Métricas principais: Total, Pendentes, Aprovadas, Em Atraso.
 - Métricas detalhadas: Finalizadas, Em Andamento, SLA Cumprido e Taxa SLA.
 - Distribuição por Etapa: usa `ETAPAS_PROCESSO`.
 - Performance por Departamento e Prioridade, e lista de “SLA em Risco”.
 - Código: `app.py` (linhas ~1234-1460). Observação: a seção de distribuição por etapa foi ajustada para renderizar as métricas em linhas de até 4 colunas (evitando erros de índice).
- ■ Histórico por Etapa
 - Filtros por Etapa, Departamento, Prioridade.
 - Tabela resultante com campos principais e botão de download CSV.
 - Código: `app.py` (linhas ~1462-1534).
- ■■ Configurações SLA
 - Exibe tabela com `SLA_PADRAO` (dias por prioridade).
 - Observação: página de leitura; ajustes não são feitos aqui (por enquanto).
 - Código: `app.py` (linhas ~1538-1557).
- ■ Gerenciar Usuários (Admin)
 - Criar usuário: Usuário*, Perfil*, Senha* (confirmação), Nome e Departamento.
 - Mensagens de erro: senhas não conferem; usuário já existe (retorna “Usuário já existe.”).
 - Listagem de usuários.
 - Redefinir senha: seleciona usuário, define nova senha (confirmação).
 - Acesso: restrito a Admin; outros perfis veem “Acesso restrito ao Admin.”.
 - Código: `app.py` (linhas ~1558-1613), criação em `add_user()` (linhas ~256-273).

Testes de Notificações

- Lado do perfil logado, o sidebar mostra “■ Notificações” com até 5 itens:
- Quando entra em “Aguardando Aprovação”: notifica “Gerência&Diretoria;”.
- Ao aprovar: notifica “Suprimentos”.
- Ao reprovar: notifica “Solicitante”.
- Código: `add_notification()` (linhas ~199-213), uso nas transições (linhas ~1103-1106, ~1126-1131, ~1219-1225) e exibição no sidebar (linhas ~644-652).

Nota: atualmente não há lógica para marcar notificações como lidas; elas tendem a permanecer visíveis.

Acesso por Perfil (Checklist Rápido)

- Solicitante:
- Tem: `■`, `■`, `■`
- Não tem: `■`, `■`, `■■`, `■`
- Suprimentos:
- Tem: `■`, `■`, `■`
- Não tem: `■`, `■`, `■■`, `■`
- Gerência&Diretoria,:
- Tem: `■`, `■`, `■`
- Não tem: `■`, `■`, `■■`, `■`
- Admin:
- Tem todas.

Casos Negativos e Validações

- Criar solicitação sem campos obrigatórios → mensagem de aviso “■■ Campos Obrigatórios”.
- Em cotação: sem valor > 0 → cotação ignorada.
- Aprovar/Reprovar sem seleção → não habilita ação.
- Mover para “Aguardando Aprovação” sem cotações válidas → verifique se pelo menos uma cotação foi registrada (ideal 3).
- Acesso não permitido:
- Tentar abrir `■ Aprovações` sem ser Gerência&Diretoria; ou Admin → “Esta página é restrita...”.
- `■ Gerenciar Usuários` sem ser Admin → “Acesso restrito ao Admin.”.

Observações Importantes (para validação e registro de issues)

- Em `■ Dashboard SLA`, confira:
- Status final usado para filtros é `Pedido Finalizado` (padronizado no código).
- Cálculo de “Pendentes” e “Em Atraso” usa os campos existentes: `carimbo_data_hora` e `sla_dias` com `calcular_dias_uteis(...)`.

Reset de Dados (opcional, ambiente local)

- Para resetar dados de teste (local), apague o arquivo de dados local do app (se aplicável) com o app fechado. Atenção: isso apaga todas as solicitações/usuários (exceto o admin que será recriado).

Resumo

- Guia de testes completo, com cenários E2E, checagens por perfil e por página, validações negativas e pontos de atenção baseados no código atual (`app.py`).
- Para dúvidas ou ajustes, entre em contato indicando o trecho e a tela onde ocorreu o problema.