

Diretrizes para o Lançamento de Planos de Projetos

Navin Ruas

Jamil Matos

2023-02-05

Maximizando a Colaboração e Produtividade: Integração do SharePoint e SiSGP

Diretrizes para o Lançamento de Planos de Projetos

- Verificações Diárias.

Verificação de Plano de Trabalho de Servidor

Objetivo

Verificar se os servidores possuem um plano de trabalho válido e enviar notificações caso não possuam.

Condições para um plano de trabalho não coberto

SituacaoPactoTrabalho = '' 'Rejeitado' 'Rascunho' 'Executado'.

Ações a serem tomadas

1. Caso o servidor não tenha um plano de trabalho válido, enviar um aviso ao servidor.
2. Se após 2 dias do primeiro aviso o servidor ainda não tiver um plano de trabalho válido, enviar um segundo aviso ao servidor e ao supervisor.

Passos

1. Obter a lista de servidores do banco SQL Portalina usando a função `pontalina(query)` do arquivo `Conexao.py`.
2. Obter a lista de férias do banco SQL Auditor usando a função `auditoria(query)` do arquivo `Conexao.py`.
3. Para cada servidor na lista de servidores:
4. Verificar se o servidor está na lista de férias. Se estiver, ignorar e passar para o próximo servidor.
5. Verificar o status `SituacaoPactoTrabalho` do servidor no banco SQL Portalina usando a função `pontalina(query)` do arquivo `Conexao.py`.
6. Se o status `SituacaoPactoTrabalho` for '' 'Rejeitado' 'Rascunho' 'Executado':
 1. Verificar se o servidor já foi notificado anteriormente (verificando em um arquivo `notificado.json`).
 2. Se o servidor não foi notificado anteriormente, enviar o primeiro aviso ao servidor usando a função `enviar_notificacao(servidor, mensagem)` do arquivo `Email.py` e registrar no arquivo `notificado.json` que o servidor foi notificado uma vez.
 3. Se o servidor já foi notificado uma vez e já se passaram 2 dias desde o primeiro aviso, enviar o segundo aviso ao servidor e ao supervisor usando as funções `enviar_notificacao(servidor, mensagem)` e `enviar_notificacao_supervisor(servidor, mensagem)` do arquivo `Email.py` e registrar no arquivo `notificado.json` que o servidor foi notificado duas vezes.

Observações

- As conexões com os bancos de dados Portalina e Auditor devem ser implementadas em um arquivo local chamado `Conexao.py`, que possui as funções `pontalina(query)` e `auditoria(query)`.
- As funções para enviar as notificações por email devem ser implementadas em um arquivo externo chamado `Email.py`.
- As mensagens de notificação estarão também em arquivos externos `avisoNCob1.html` e `avisoNCob2.html`.

Validar Conclusão do Plano de Trabalho

Objetivo

Verificar se os servidores concluíram todas as atividades do plano de trabalho dentro do prazo e enviar notificações caso não tenham.

Ações a serem tomadas

1. Caso o plano de trabalho não esteja com todas as atividades concluídas até o prazo, enviar um aviso ao servidor.

Passos

1. Obter dados do SQL usando a consulta `SELECT *`.
2. Separar os dados por servidor e verificar o status mais recente de `SituacaoPactoTrabalho` para cada servidor.
3. Para cada servidor:
4. Se todas as atividades com o mesmo `pactoTrabalhoId` tiverem status `Concluída`, ignorar e passar para o próximo servidor.
5. Se a data `DtFimPactoTrab` estiver faltando 1 dia para vencer, enviar uma notificação ao servidor usando o arquivo `HTML avisoConc1.html`.
6. Se a data `DtFimPactoTrab` já estiver vencida, enviar uma notificação ao servidor e ao supervisor usando o arquivo `HTML avisoNConc.html`.
7. Adicionar o servidor à lista de servidores não concluídos em um arquivo `nConc.json`.
8. Verificar se os servidores na lista já concluíram as atividades. Caso tenham concluído, remover da lista.

Observações

- As conexões com o banco de dados SQL devem ser implementadas em um arquivo local chamado `Conexao.py`, que possui uma função para executar consultas SQL.
- As funções para enviar as notificações por email devem ser implementadas em um arquivo externo chamado `Email.py`.
- As mensagens de notificação estarão também em arquivos externos `avisoConc1.html` e `avisoNConc.html`.

Verificar Campo Descrição

Objetivo

Verificar se o campo `Descrição` está correto e coincide com a atividade quando o trabalho está 'Enviado para Aceite'. Caso haja algum erro, apontar o(s) erro(s) ao servidor e ao supervisor.

Passos

1. Obter a lista de trabalhos com status 'Enviado para Aceite' do banco de dados usando uma consulta SQL.
2. Para cada trabalho na lista:
3. Verificar se o campo Descrição está correto e coincide com a atividade.
4. Se houver algum erro, enviar uma notificação ao servidor e ao supervisor apontando o(s) erro(s) usando a função do arquivo Email.py com mensagem descErro.html.
5. Se tudo estiver correto, enviar o trabalho para o supervisor para análise com a mensagem descAceita.html.

Observações

As conexões com o banco de dados devem ser implementadas em um arquivo local chamado Conexao.py, que possui uma função para executar consultas SQL. As funções para enviar as notificações por email devem ser implementadas em um arquivo externo chamado Email.py.

Verificar existencia do produto no eAud

- Atividade Avaliação ou Consultoria.
- Se não houver arquivo na id do eAud.
- No dia seguinte avisar supervisor e servidor novamente.
- Se concluído atividades e produtos avisar supervisor.

Atualizar Gerador de Descrição

Atualização do Dashboard Gerencial SharePoint

- 'Em Execução' ou 'Autorizado' enviar dados para SharePoint.