

"

Inicialização (Logger de Sessão) GSE-LLR-159

A interface de inicialização do controlador deve instanciar o GseLogger e armazená-lo para uso futuro.

Inicialização (Log de Caminho) GSE-LLR-161

A interface de inicialização do controlador deve logar o caminho do arquivo de log (obtido da interface do GseLogger).

"

Procedimento Realizado

Metodologia de Análise estática de código

Análise no arquivo [gse/backend/controllers/upload_controller.py](#)

Resultados obtidos

Os requisitos estão rastreados e estão implementados através do método construtor da classe UploadController do arquivo referido no procedimento

```
85      """
86      # REQ: GSE-LLR-159: Inicialização (Logger de Sessão)
87      # Descrição: A interface de inicialização do controlador DEVE instanciar o
88      #   `GseLogger` e armazená-lo para uso futuro.
89      # ---
90      # REQ: GSE-LLR-160: Inicialização (Log de Início de Sessão)
91      # Descrição: A interface de inicialização do controlador DEVE invocar o
92      #   handler de log (GSE-LLR-162) com uma mensagem "SESSÃO GSE INICIADA".
93      # ---
94      # REQ: GSE-LLR-161: Inicialização (Log de Caminho)
95      # Descrição: A interface de inicialização do controlador DEVE logar o
96      #   caminho do arquivo de log (obtido da interface do `GseLogger`).
97      #
98      # Autor: Julia
99      # Revisor: Fabrício
100     # =====
101     def __init__(self, parent=None):
102         """
103             Implementa: GSE-LLR-158, 159, 160, 161
104         """
105         super().__init__(parent)
106         # GSE-LLR-158
107         self.threadpool = QThreadPool()
108
109         # GSE-LLR-159
110         self.file_logger = GseLogger()
111
112         self.selected_path = ""
113         self.selected_pn = ""
114
115         # GSE-LLR-160
116         self._log_handler(f"--- SESSÃO GSE INICIADA ---")
117         self._log_handler(
118             f"Controlador de Upload inicializado. Threads: {self.threadpool.maxThreadCount()}"
119         )
120         # GSE-LLR-161
121         self._log_handler(f"Log de sessão salvo em: {self.file_logger.get_log_path()}")
```