

N2 INTERDISCIPLINAR

Grupo de 3 a 5 integrantes.

Esta atividade valerá como (N2) do 2º bimestre nas disciplinas de Engenharia de Software, Estatística e Pesquisa Operacional, conforme as entregas específicas de cada disciplina descritas neste documento.

40% da pontuação será atribuída à apresentação (seminário), com duração de até 10 minutos, destacando os principais pontos do projeto em cada disciplina. A apresentação deverá ocorrer no dia da avaliação da respectiva disciplina, conforme o calendário acadêmico da faculdade.

20% da pontuação será atribuída conforme **A Avaliação de Pares dentro dos grupos.**

Atenção: Poderá ser entregue um único pdf para todas as disciplinas ou arquivos separados.

Objetivo: Desenvolver uma aplicação em **linguagem Python que simule o comportamento de filas com múltiplos servidores (modelo M/M/c), utilizando dados reais ou simulados (em CSV), calcular métricas estatísticas e aplicar princípios ágeis de desenvolvimento com base no framework Scrum.**

Uma organização (ex: clínica, restaurante, central de suporte) deseja entender o comportamento do seu atendimento frente à demanda. Para isso, disponibilizou dados sobre o tempo entre chegadas dos clientes e o tempo que cada atendimento levou. Seu grupo será responsável por simular esse sistema e, com base nos resultados, realizar uma análise estatística dos dados reais e dos gerados.

O grupo deve escolher **1 entre os 3 contextos abaixo**:

- 1. Clínica de vacinação**
- 2. Restaurante universitário**
- 3. Central de suporte técnico**

PROGRAMAÇÃO/ PESQUISA OPERACIONAL

1. Ler um arquivo CSV com dados de tempo entre chegadas e tempo de atendimento
2. Simular o funcionamento de um sistema de filas M/M/c
3. Calcular as seguintes métricas:
 - P_0 : Probabilidade do sistema estar vazio
 - P_{espera} : Probabilidade de espera
 - L_q : Número médio na fila
 - W_q : Tempo médio de espera
 - W : Tempo médio no sistema
 - L : Número médio no sistema
4. Visualização do comportamento da fila
Gráfico de:
 - **Tempo de espera por cliente**
 - **Tamanho da fila ao longo do tempo**
 - **Tempo de ocupação dos servidores**
5. Apresentar um relatório técnico e visualizações com as métricas.

Entrega da parte Pesquisa Operacional:

- **Código .py**
- **resultados.csv gerado pela simulação**
- **Relatório (pdf):**
 - **Contendo os resultados dos itens 3, 4 .**
 - **Responder as seguintes perguntas:**
 - **Vale a pena adicionar mais um servidor?**
 - **Qual seria o impacto de um atendente mais rápido (μ maior)?**

ESTATÍSTICA

1. Calcular estatísticas descritivas sobre os dados do CSV:
 - Média, mediana, moda, variância, desvio padrão
2. Visualizar os dados:

- Histograma para:
 - Tempo de atendimento
 - Tempo de espera
- Boxplot comparando:
 - Tempos de atendimento vs tempos de espera

3. Comparações e Inferência:

- Apresente o intervalo de confiança:
 - Para a média do tempo de atendimento
 - Para a média do tempo de espera

Entrega da parte Estatística:

- **Código de análise .py ou .ipynb**

Relatório: relatorio_estatistico.pdf, com:

- **Os resultados dos itens de 1 a 3**
- **Interpretação dos dados e gráficos.**
- **Sugestão de melhorias no atendimento com base nos dados**

ENGENHARIA DE SOFTWARE

1. Papéis Scrum :

- Product Owner: responsável por manter o backlog
- Scrum Master: guia o grupo nos ritos Scrum
- Time de Desenvolvimento: responsável por implementar o projeto

2. Product Backlog com pelo menos 10 itens

Como por exemplo:

ID	Descrição	Prioridade	Sprint
PB1	Ler o CSV com pandas	Alta	1
PB2	Simular fila com múltiplos servidores	Alta	1
PB3	Calcular métricas da fila	Alta	1
PB4	Gerar gráficos da simulação	Média	2

ID	Descrição	Prioridade	Sprint
PB5	Analisar estatisticamente os tempos de chegada	Alta	2
PB6	Criar boxplot dos tempos de espera	Média	2
PB7	Validar distribuição de Poisson	Baixa	3
PB8	Criar relatório final da simulação	Alta	3
PB9	Criar protótipo do dashboard (Streamlit)	Média	3
PB10	Fazer retrospectiva final do projeto	Média	3

3. Planejamento por Sprint

- Sprint 1: base funcional da simulação
- Sprint 2: análise estatística e gráficos
- Sprint 3: interface, ajustes, finalização e apresentação

4. Quadro Kanban (Trello, Planner, etc.)

- Use colunas: Backlog, A Fazer, Em Progresso, Concluído
- Print da tela das tarefas

5. Cerimônias Scrum (simuladas):

- Sprint Planning (definição de tarefas por sprint)
- Daily Scrum (registro semanal de progresso)
- Sprint Review (entrega ao final de cada sprint)
- Retrospective (análise do processo e aprendizados)

ACOMPANHAMENTO COM GIT

Repositório Git /GitHub contendo:

- Código das simulações (Python)
- Código das análises estatísticas
- Documentos (backlog, relatórios, gráficos)

2. Histórico de commits organizado:

- Commits com mensagens claras e contextualizadas
- Commits separados por tarefa ou componente
- Uso de branches é recomendado (ex: simulacao, estatistica, interface)

3. Registro do uso do Git:

- Print da **tela de commits**.
- Print da **estrutura do repositório**
- Print de merges e branches

4. Sugestão de README.md do repositório:

- Objetivo do projeto
- Instruções de execução
- Divisão de tarefas (por disciplina)
- Referências

Entrega da PARTE Engenharia de Software:

Relatório em pdf com todos os prints do GIT e das etapas SCRUM.