

Lab Exame de CSC-27_CE-288 / 2024

CTA - ITA - IEC

Prof Hirata e Prof Juliana

Objetivo: Desenvolver uma **aplicação com desafio na área de Sistemas Distribuídos**.

- Equipe **de até 4 alunos com participação efetiva em Sistemas Distribuídos**.
Obs: Não vale o aluno ser apenas de *front* ou relator.
- A aplicação deve ser completa e testada (sem erros).
- A aplicação deve ser concebida pela equipe. Não é permitido utilizar aplicações prontas. Pesquisem bem para terem boas ideias!
- **DC sempre!** Muito cuidado com uso de ChatGPT (e similares): pode usar para apoiar na compreensão dos tópicos/algoritmos e para dúvidas de implementação. Não deve ser usado para realizar o projeto por você!
- Dependendo da qualidade do projeto, podemos posteriormente trabalhar num artigo para submeter a uma conferência internacional.
- Dependendo da qualidade da proposta, podemos desenvolver mais para definir um TG.

Requisitos do projeto:

Você escolhe o enfoque do seu trabalho!

- É recomendado o uso de Docker para facilitar instalação/utilização.

Possíveis abordagens/assuntos:

- Blockchain: desenvolver uma aplicação para algum problema particular.
 - Ex: usar outras plataformas (como Hyperledger), ou explorar conceitos como *bridges*, NFT (*non-fungible token*) e *oracles*.
- Um trabalho ligado à aplicação de tecnologia a um caso interessante.
 - Ex: Hadoop MapReduce, Spark ou outras.
- Um projeto tratando de tópicos não abordados na disciplina.
 - *Distributed Messaging System*. Ex: implementar em Go serviços de *queue* e *publish-subscribe* baseando-se na tecnologia RabbitMQ, e usar em uma aplicação.
 - *Distributed Hash Table*. Ex: implementar em Go o algoritmo Pastry, e usar em uma aplicação.
 - *Kubernetes*. Ex: atividades/aplicações práticas sobre gerenciar/escalonar *containers*, e reiniciar/substituir *containers* falhos.
 - *Distributed File System (DFS)*. Ex: implementar prática(s) para entender DFS, criar/gerenciar, realizar operações CRUD (*Create, Read, Update, Delete*) e demonstrar tolerância a falhas, baseando-se em tecnologias completas (como HDFS e Ceph) ou mais simples (como MooseFS e GlusterFS).
 - *Distributed Query Processing*. Ex: implementar em Go técnicas/algoritmos como Google Dremel.
- Verificação formal de algoritmos distribuídos
 - Ex: usando ferramenta de prova de teorema (como Isabelle) ou de model-checking (como SPIN com linguagem Promela).
- Metaverso (realidade virtual e aumentada) e Sistemas Distribuídos, a ver:
 - Aplicação distribuída com óculos e tablet para alguma necessidade.
 - Aplicação em Metaverso integrada com Blockchain (ex: para compra

- de itens de jogo).
- Ensino de tecnologias/algoritmos distribuídos (como Blockchain).

Entrega Parcial

- **Documentos no Classroom**
 - **PPT parcial:** focar nos itens 1-4 do Anexo
- **Apresentação presencial (~ 5 min):** Toda equipe deve participar!

Entrega Final

- **Documentos no Classroom**
 - **PPT final:** focar nos itens 3-6 do Anexo
 - **Relatório (pdf):** todos itens do Anexo
 - **Códigos**
- **Apresentação presencial (~ 15 min).** Toda equipe deve participar!

ANEXO: TÓPICOS ESPERADOS PARA O PROJETO

- O texto pode ser feito em Português ou Inglês (indicado para quem deseja se aprofundar nesse trabalho depois).
 - 1) Motivation**
 - Explicar o cenário/tema de interesse.
 - Indicar qual problema que você deseja atacar.
 - Explicar o desafio em Sistemas Distribuídos.
 - 2) Related Work**
 - Citar o que outros autores estão fazendo sobre esse tema e o que está faltando (*gap*).
 - O ideal aqui é pesquisar artigos científicos relacionados, ou aplicações/sites/iniciativas existentes que abordem o tema.
 - 3) Objective**
 - Indicar o que será feito. Ou seja, qual proposta será desenvolvida (de forma a atacar o *gap* identificado).
 - 4) Proposal**
 - Descrever os requisitos funcionais e não funcionais.
 - Explicar a proposta. Apresentar de forma visual (com esquema/figura) o que será feito e os componentes/atores/tecnologias empregados.
 - 5) Prototype**
 - Explicar a implementação realizada.
 - Apresentar figuras que comprovem o desenvolvimento do protótipo. Ex: front-end, testes realizados, etc.
 - Discutir sobre os testes realizados.
 - Indicar como instalar/executar/rodar a aplicação.
 - 6) Conclusions**
 - Relembrar o objetivo/vantagens da proposta.
 - Fazer uma discussão geral da proposta e do protótipo, indicando desafios superados e limitações existentes.
 - Indicar trabalhos futuros.
 - 7) Group organization**
 - Indicar como cada aluno contribuiu.

Aluno	Atividades	Percentual do trabalho realizado (soma dessa coluna deve ser 100%)
<i>Aluno 1</i>		50%