

Unidade Acadêmica de Engenharia Elétrica
Centro de Engenharia Elétrica e Informática (CEEI)
Universidade Federal de Campina Grande (UFCG)

Disciplina: Informática Industrial

Equipe: Danilo Gomes de Andrade



Plano do Projeto

13/08/2021



Sumário

1. Visão Geral do Projeto	2
2. Desenvolvimento	2
3. Product Backlog	3
4. Validação e Verificação	7
4.1. Testes de desenvolvimento	7
4.2. Testes de aceitação	8



1. Visão Geral do Projeto

O objetivo geral do sistema é acompanhar e monitorar níveis de óleos e quantidade de gases produzidos a fim de assegurar as obtenções dos produtos desejados a partir do petróleo. Dessa forma, pode-se auxiliar a indústria petrolífera no processo da destilação fracionada com o acompanhamento e a previsão da quantidade de cada componente que deve ser obtida utilizando inteligência artificial, podendo rastrear algum erro na produção devido a vazamentos identificados caso haja uma inconsistência observada e, conseqüentemente, levando à economia de maiores gastos devido a eventuais falhas.

A utilização de mensagens de avisos no aplicativo indicando se os dados estão em conformidade ou não, sua gravidade e a interação com o usuário para confirmar as ações desejadas visam à implementação eficiente do sistema, já que pode-se utilizar o aplicativo móvel de qualquer lugar a qualquer hora, analisando e tomando as medidas necessárias para evitar maiores perdas ou acidentes de acordo com o usuário cadastrado.

Para tanto, uma interface em aplicativo móvel deve ser desenvolvida em conjunto com o sistema de aquisição de dados, uma API, uma inteligência artificial perceptron multicamadas (MLP) e o armazenamento em nuvem do banco de dados, garantindo a consulta em tempo real de diversos locais.

Contudo, observa-se que, devido à falta de acesso dos integrantes da equipe aos reservatórios de óleos e fases na indústria petrolífera, ver o projeto monitorando e atuando com os dados reais e precisos torna-se inviável, bem como devido à falta de precisão dos equipamentos utilizados. Além disso, não se sabe com exatidão a fração de cada parte que é obtida do petróleo para conseguir montar o cálculo do sistema.

Levando em consideração todos os pontos apontados do sistema, há uma previsão para sua conclusão em 5 meses, sendo possível estender esse prazo caso alguma incrementação seja desejada.

2. Desenvolvimento

Para implementação do sistema, alguns processos de desenvolvimento serão adotados de forma a agilizar o projeto. Ou seja, enquanto os requisitos iniciais são feitos, novos são criados caso haja necessidade e, em conjunto, testes são feitos em paralelo, de forma a verificar o funcionamento de cada parte do sistema. Dentre os processos de desenvolvimento, vale salientar o *user stories* (advindo do Extreme Programming XP), visto que os usuários indicam funcionalidades desejadas, auxiliando no esclarecimento dos requisitos, e que o time deve identificar melhorias necessárias ou não. Para tanto, o foco será no funcionamento do sistema, com documentação simples e sucinta. Dessa forma, a gestão será feita utilizando o SCRUM pelo [Taiga.io](https://taiga.io). Além disso, o projeto será desenvolvido utilizando o [github](https://github.com).

Com relação ao projeto, alguns atores do processo devem ser ressaltados:

- Product Owner: Danilo Freire de Souza Santos.
- Scrum Master: Joãozinho Silva Ferreira.
- Scrum Team: Danilo Gomes de Andrade.



Assim, as sprints terão duas semanas de duração, com reuniões diárias sucintas. A cada término de sprint, as próximas duas semanas são reorganizadas.

Logo, com relação às entregas do projeto, pode-se listar:

- 27/08: Banco de dados funcionando com a API
- 10/09: Sistema de controle e de aquisição de dados
- 24/09: CRUD de usuários funcionando
- 08/10: Inteligência Artificial treinada
- 22/10: Interface Mobile funcionando para acessar usuário
- 05/11: Interface Mobile com controle das válvulas
- 19/11: Interface Mobile com exibição dos gráficos e com sistema de notificação

3. Product Backlog

ID	Título	Descrição	Esforço
DMPB01	Desenvolvimento do Banco de Dados SQL em nuvem (Azure)	Descrição: Implementar o banco de dados no Azure e organizar as tabelas que serão utilizadas no projeto (Armazenar os valores dos sensores e os dados dos usuários) Requisitos associados: DM02, DM05, DM07, DM09, DM17. Critério de Aceitação: Banco de Dados SQL Azure configurado com as tabelas.	3
DMPB02	Desenvolvimento inicial API em Python	Descrição: Desenvolver a API em python para conseguir adquirir dados de usuário e valores de sensores. Requisitos associados: DM02, DM07, DM19. Critério de Aceitação: API conseguindo receber os valores.	5
DMPB03	Conexão entre API e o Banco de Dados	Descrição: Conectar API de forma a conseguir armazenar ou buscar dados no banco de dados. Requisitos associados: DM02, DM05, DM07, DM09, DM17.	2



		Critério de Aceitação: Armazenar e ler valores no banco de dados pela API.	
DMPB04	Montagem do Arduino com Sensores e Atuadores	Descrição: Conectar sensores e atuadores ao arduino. Requisitos associados: DM01, DM02, DM14, DM16. Critério de Aceitação: Adquirir valores dos sensores e controlar os atuadores.	3
DMPB05	Conexão do Arduino ao programa em Python	Descrição: Conectar o arduino de forma serial a fim de armazenar os valores lidos no programa em python e controlar o atuador. Requisitos associados: DM01, DM02, DM14, DM16, DM19. Critério de Aceitação: Obter valores do arduino no programa principal em python, por onde se pode controlar o atuador.	5
DMPB06	Conexão dos Sensores ao banco de dados	Descrição: Salvar valores lidos pelos sensores diretamente no banco de dados ao longo do tempo utilizando a API. Requisitos associados: DM01, DM02, DM14, DM16, DM19. Critério de Aceitação: Salvar no banco de dados os valores lidos pelos sensores.	2
DMPB07	Configuração do CRUD de usuário em Python	Descrição: Configurar o sistema de segurança para os usuários de forma a criptografar os dados sensíveis. Requisitos associados: DM05, DM22, DM12, DM15, DM17, DM19. Critério de Aceitação: Conseguir criptografar os dados desejados no banco de dados e configurar o CRUD de usuário.	8



DMPB08	Configurar Inteligência Artificial	Descrição: Configurar o sistema de Inteligência Artificial em Python utilizando a biblioteca do scikit-learn para usar MLP (perceptron multicamadas) Requisitos associados: DM03, DM08, DM09, DM21, DM20. Critério de Aceitação: Conseguir gerar a MLP.	8
DMPB09	Treinar a Inteligência Artificial	Descrição: Adquirir dados e utilizá-los para treinar a MLP. Requisitos associados: DM03, DM08, DM09, DM21, DM20. Critério de Aceitação: Inserir um dado e obter a resposta com taxa de erro abaixo de 0.01%	2
DMPB10	Gerar Interface homem-máquina básica	Descrição: Gerar modelo inicial da interface homem-máquina utilizando React Native e conectá-lo à API. Requisitos associados: DM02, DM03, DM04, DM06, DM07, DM10, DM11, DM13, DM18. Critério de Aceitação: Gerar o modelo inicial	8
DMPB11	CRUD de usuários na interface	Descrição: Criar, ler, atualizar e deletar usuários pela interface. Requisitos associados: DM02, DM03, DM04, DM05, DM06, DM07, DM22, DM10, DM11, DM12, DM13, DM18, DM15, DM17, DM19. Critério de Aceitação: Conseguir gerar um usuário e o utilizar pela interface.	5



DMPB12	Controle dos atuadores pela interface	Descrição: Utilizar o controle de atuadores pela interface utilizando a API. Requisitos associados: DM01, DM06, DM11, DM12 . Critério de Aceitação: Conseguir abrir ou fechar uma válvula pela interface.	5
DMPB13	Obtenção dos dados dos sensores na interface	Descrição: Uso da API para obter do banco de dados os valores dos sensores de determinado período especificado na interface (filtrar por tipo de produto). Requisitos associados: DM02, DM03, DM07, DM17. Critério de Aceitação: Conseguir selecionar um período e obter os dados dos níveis do banco de dados.	5
DMPB14	Gerar gráficos na interface	Descrição: Gerar gráfico a partir dos dados obtidos em DMPB13. Requisitos associados: DM02, DM03, DM07, DM17. Critério de Aceitação: Conseguir gerar um gráfico dos níveis de um período na interface.	8
DMPB15	Gerar gráfico ideal na interface	Descrição: Gerar gráfico a partir dos dados obtidos da inteligência artificial da previsão para os valores de cada produto. Requisitos associados: DM03, DM07, DM08, DM21, DM17, DM20. Critério de Aceitação: Conseguir gerar um gráfico dos níveis de um período na interface.	5



DMPB16	Implementar sistema de notificações no celular	Descrição: Receber notificações sempre que houver alguma inconsistência identificada pela Inteligência Artificial. Requisitos associados: DM04, DM09, DM10, DM11, DM18. Critério de Aceitação: Receber notificações do estado do sistema no celular.	5
---------------	---	--	----------

4. Validação e Verificação

Para a validação e verificação do sistema, alguns testes devem ser efetuados. Durante o desenvolvimento, testes de unidade devem identificar a funcionalidade dos objetos, testes de componente devem identificar a funcionalidade das interfaces entre os componentes e testes de sistema devem identificar a funcionalidade do todo. Com o projeto completo, testes de *release* são utilizados para identificar a aceitação do sistema e a disponibilização para o usuário. Dessa forma, os testes de usuário devem encontrar alterações pela visão do seu ambiente, sendo alpha quando testam junto com os desenvolvedores num ambiente controlado, no beta uma versão é liberada para experimentar os ambientes e a aceitação é para saber se o sistema está pronto para ser lançado.

Para tanto, a ferramenta Testlink deve ser utilizada visando ao gerenciamento de testes. Além disso, o framework Selenium será utilizado para capturar dados biométricos do usuário e verificar as asserções do sistema.

4.1. Testes de desenvolvimento

Com relação aos testes de unidade, de componente e de sistema, pode-se listar:

ID DMT01	
Título	<i>Teste de integração do banco de dados</i>
Sumário	<i>Deve-se salvar valores nas tabelas do banco de dados e lê-los em seguida utilizando a API.</i>
Requisitos Associados	DM02, DM05, DM07, DM09, DM17.
Pré-condições	<i>Banco de dados configurado.</i>
Critério de Sucesso ou Aceitação	<i>Conseguir salvar e ler todos valores com sucesso.</i>



ID DMT02	
Título	<i>Teste de hardware dos sensores e atuadores do arduino</i>
Sumário	<i>Deve-se ler valores de cada sensor e ativar/desativar cada atuador a partir do arduino.</i>
Requisitos Associados	DM01, DM02, DM14, DM16, DM19.
Pré-condições	<i>Hardware montado.</i>
Critério de Sucesso ou Aceitação	<i>Conseguir ler os valores e controlar os atuadores com sucesso.</i>

ID DMT03	
Título	<i>Teste de integração dos sensores e atuadores do arduino</i>
Sumário	<i>Deve-se enviar valores de estado dos atuadores para o arduino e ler valores dos sensores pela comunicação serial com o programa principal desenvolvido em Python.</i>
Requisitos Associados	DM01, DM02, DM14, DM16, DM19.
Pré-condições	<i>Hardware montado e teste DMT04.</i>
Critério de Sucesso ou Aceitação	<i>Conseguir ler os valores e controlar os atuadores com sucesso.</i>

ID DMT04	
Título	<i>Teste de CRUD de usuário</i>
Sumário	<i>Tentar criar uma conta, editar, observar os níveis de acesso, excluir, acessar e sair dela.</i>
Requisitos Associados	DM05, DM22, DM12, DM15, DM17, DM19.
Pré-condições	<i>Sistema de usuários montado.</i>
Critério de Sucesso ou Aceitação	<i>Conseguir realizar todas as atividades com testes para usuários repetidos.</i>

ID DMT05	
Título	<i>Teste de qualidade da inteligência artificial</i>
Sumário	<i>Pegar os dados reais e verificar a taxa de acerto da inteligência artificial.</i>
Requisitos Associados	DM03, DM08, DM09, DM21, DM20.
Pré-condições	<i>Inteligência artificial treinada.</i>
Critério de Sucesso ou Aceitação	<i>Obter uma taxa de acerto acima de 99,99%.</i>

4.2. Testes de aceitação

Com relação aos testes de aceitação, pode-se listar:



ID	DMT06
Título	<i>Teste de CRUD de usuário utilizando a interface</i>
Sumário	<i>Criar, ler, atualizar e deletar usuários utilizando a interface mobile.</i>
Requisitos	DM05, DM22, DM12, DM15, DM17, DM19.
Associados	
Pré-condições	<i>DTM04.</i>
Critério de Sucesso ou Aceitação	<i>Conseguir realizar todas as atividades com testes para usuários repetidos.</i>

ID	DMT07
Título	<i>Teste de integração das válvulas com a interface</i>
Sumário	<i>Conseguir controlar as válvulas utilizando a interface levando em consideração o nível de acesso do usuário.</i>
Requisitos	DM01, DM06, DM11, DM12 .
Associados	
Pré-condições	<i>DTM02, DTM03, DTM04.</i>
Critério de Sucesso ou Aceitação	<i>Abrir/fechar válvulas utilizando a interface.</i>

ID	DMT08
Título	<i>Teste de funcionalidade dos gráficos na interface</i>
Sumário	<i>Testar funcionalidades do gráfico como alteração de período, seleção de dados e alteração de escala.</i>
Requisitos	DM02, DM03, DM07, DM08, DM21, DM17, DM20.
Associados	
Pré-condições	<i>DTM01.</i>
Critério de Sucesso ou Aceitação	<i>Conseguir realizar todas as funções dos gráficos.</i>

ID	DMT09
Título	<i>Teste de exibição de alertas na interface</i>
Sumário	<i>Simular erros para verificar o funcionamento das sinalizações deles.</i>
Requisitos	DM04, DM09, DM10, DM11, DM18.
Associados	
Pré-condições	-
Critério de Sucesso ou Aceitação	<i>Receber notificações indicando quando há uma discrepância acima de 5% ou 10% com relação ao esperado e indicar qual medida deveria ser/foi tomada.</i>



ID	DMT10
Título	<i>Teste de previsão de produção</i>
Sumário	<i>Utilizar a inteligência artificial para prever determinada produção e comparar com os resultados obtidos.</i>
Requisitos Associados	DM03, DM08, DM09, DM21, DM20.
Pré-condições	<i>Inteligência artificial treinada</i>
Critério de Sucesso ou Aceitação	<i>Encontrar uma taxa de acerto acima de 99,99% para a MLP.</i>