

Atividade de programação I – desenvolvimento de sistemas

Todas as interfaces devem ser construídas utilizando a API flet <https://flet.dev/>

Se necessário, você pode utilizar outras APIS.

Tema 1: Jogo acerte o número e adição e subtração

Objetivo: Trata-se do desenvolvimento de um jogo de duas fases. Na primeira fase o programa deve sortear um número maior que zero. O usuário tentará acertar este número. Para cada tentativa errada, o programa deve informar se o próximo chute deve ser um número maior ou menor. Na segunda fase, o usuário terá um minuto para realizar o máximo de operações de adição e subtração possíveis.

Requisitos:

- 1- NaAo abrir o jogo o usuário deve informar um nickname começando com @
- 2- Após informar corretamente o nickname o usuário deve ir para a primeira fase do jogo
- 3- Na primeira fase, o usuário deve visualizar o número de tentativas já realizadas.
- 4- Na segunda fase do jogo o usuário terá um minuto para realizar o máximo de operações possíveis.
- 5- Mesmo que não tenha acertado, o programa deve sortear a próxima operação aritmética.
- 6- O sorteio da operação aritmética deve sortear os dois números, bem como o operador.
- 7- Ao final da segunda fase, o usuário deve visualizar quantas operações ele realizou corretamente.
- 8- O programa deve ter um Menu com duas opções: Jogar, Ranking.
- 9- Na tela de ranking o usuário deverá visualizar a classificação dos jogadores. Deve-se exibir o ranking da primeira fase e da segunda fase separadamente.
- 10- As informações do ranking devem ser armazenadas em um arquivo.
- 11- Se o usuário jogar várias vezes com o mesmo nickname, então o ranking só deve ser atualizado se ele obtiver uma pontuação maior.

Tema 2: Sistema dashboard

Objetivo: Trata-se do desenvolvimento de um sistema de visualização da informação. Considere como fonte de dados a base de dados de acidentes da PRF que foi disponibilizada no SIGAA.

Requisitos:

1. Devem ser criados dez gráficos, considerando as questões da lista de exercício de manipulação de arquivos.
2. Cada gráfico deve ser exibido em uma aba do programa.
3. Para cada aba crie opções de filtro a depender do contexto da questão. Por exemplo, se o gráfico está mostrando a frequência de acidentes por UF, então crie componentes visuais que permita filtrar quais UFs eu queira analisar e exibir no gráfico.

Tema 3: Sistema de criptografia

Objetivo: Trata-se de um sistema para prover a confidencialidade de mensagens ou arquivos.

Requisitos:

1. O programa deve ter um Menu com duas opções: cifrar e decifrar.
2. Na tela de cifragem o usuário irá escolher se deseja cifrar uma mensagem ou um arquivo.

3. Para cifrar mensagem o usuário deve digitar a mensagem e uma frase senha que será utilizada para gerar a chave secreta.
4. O processo de cifragem deve ser realizado utilizando o algoritmo AES.
5. A mensagem cifrada deve ser exibida na mesma área de texto em que foi digitada a mensagem original.
6. Para cifrar arquivo o usuário deve fazer o upload e informar a frase senha que será utilizada para gerar a chave secreta.
7. O arquivo cifrado deve ter a extensão .cripto e deve ser armazenado no mesmo diretório do arquivo original.
8. O processo de decifragem, tanto de arquivo quanto de mensagem, é similar ao processo de cifragem. A única diferença é que a mensagem e arquivo devem corresponder ao conteúdo cifrado.

Tema 4: Sistema de extração de conteúdo

Objetivo: Trata-se de um programa de extração de dados com expressão regular.

Requisitos:

1. Inicialmente o usuário deve informar a pasta em que estão os arquivos os quais deseja extrair dados.
2. O usuário deve selecionar os dados que deseja extrair. As opções são as seguintes: CPF, Telefone, E-mail, Nome próprio.
3. A extração deve ser realizada por expressão regular ou lista de referência. Seja criativo!
4. Após a extração o programa deve salvar as informações extraídas em um arquivo CSV e deve informar na tela a quantidade de dados extraídos por categoria.

Tema 5: Sistema de controle de registros

Objetivo: Trata-se de um sistema de cadastro de usuário.

1. Na tela de cadastro o usuário deve informar os seguintes dados: Nome, Telefone, CPF, RG, Endereço, Nascimento, E-mail, upload de Foto.
2. Você deve validar todos os campos e informar para o usuário caso algum dado informado seja inválido. Seja criativo!
3. Todos os campos são obrigatórios.
4. O cadastro só pode ser realizado quando todos os dados forem válidos.
5. Na tela de pesquisa os dados dos usuários devem ser exibidos em uma tabela e deve ser permitido o filtro por qualquer dado cadastrado.
6. Na tabela de pesquisa devem constar os botões para editar e remover cadastros.
7. Na tela de atualização as mesmas regras de validação do cadastro devem ser aplicadas.
8. Os dados dos usuários devem ser armazenados em um arquivo csv.
9. No arquivo csv, a coluna foto deve armazenar o path de onde a foto está armazenada.

Tema 6: Sistema de vendas com carrinho de compras

Objetivo: Trata-se de um sistema de simulação de vendas.

Requisitos:

1. Na tela principal deve ser um grid com os produtos disponíveis para venda.
2. Cada produto deve ser um card com a imagem do produto, descrição e valor.

3. Na barra superior deve constar o nome do sistema e no canto superior direito o carrinho de compras.
4. Quando o usuário clicar no card de um produto, deve-se adicionar este item ao carrinho.
5. A adição do produto ao carrinho implica na atualização do carrinho que deve exibir a quantidade de produtos já adicionados.
6. Se o usuário clicar N vezes no mesmo produto, então este produto deve ser adicionado N vezes ao carrinho.
7. Quando o usuário clicar no carrinho, ele vai para a tela que lista os produtos adicionados ao carrinho. Nesta tela ele pode remover produtos, pode voltar para a tela de vendas ou pode ir para a tela de pagamento.
8. Na tela de pagamento o usuário deve informar a forma de pagamento e efetuar o pagamento.

Tema 7: Sistema de anotações com calendário

Objetivo: Trata-se de um sistema estilo google calendar.

Requisitos:

1. Na tela principal o usuário deve visualizar o calendário mensal.
2. A tela principal deve ter a funcionalidade que permita inserir uma anotação.
3. Na anotação o usuário deve informar o conteúdo da nota e a data.
4. No calendário mensal, todos os dias que tiverem notas, deve ter alguma indicação visual.
5. Quando o usuário clicar em um dia do calendário mensal, deve abrir uma nova tela para listar todas as notas daquele dia.
6. Na tela que lista todas as notas de um dia, o usuário pode excluir ou atualizar alguma nota.
7. As notas devem ser armazenadas em um arquivo csv.
8. O calendário pode ser de um único mês.

Tema 8: Sistema de gerenciamento senha

Objetivo: Trata-se de um sistema para gerar senhas sintéticas fortes.

Requisitos:

1. O usuário deve criar o seu cadastro contendo somente login e senha.
2. Na tela principal o usuário deve informar o sistema para o qual deseja criar a senha
3. A senha deve ser gerada utilizando um algoritmo de hash, por exemplo o SHA3.
4. No menu deve ter a opção de gerenciar senhas
5. Na tela de gerenciamento de senhas o usuário de verificar em uma tabela todas as senhas criadas e os respectivos sistemas.
6. Nesta tabela deve conter as opções de excluir senha ou renovar senha.
7. Ao renovar a senha, uma nova senha deve ser gerada para o sistema relacionado.
8. As senhas dos sistemas devem ser salvas em um arquivo csv.

Tema 9: Sistema de simulação de aposta do brasileirão

Objetivo: Trata-se de um sistema de aposta de uma rodada do campeonato brasileiro.

Requisitos:

1. Na tela principal deve contar os confrontos de uma roda do campeonato brasileiro

2. Para cada confronto deve constar o nome do time e o seu escudo.
3. O usuário deve preencher os seus palpites para a rodada.
4. Em seguida deve informar o valor da aposta.
5. No menu, deve conter uma opção instruções.
6. Quando o usuário clicar no menu instruções, abrirá uma nova tela onde constará todas as explicações de possibilidades de aposta.
7. Na tela principal, além de preencher os placares o usuário deve informar o valor da aposta.
8. Ao clicar em apostar, o programa deve simular os placares da rodada e, em uma nova tela, mostrar o resulta simulado e o resultado apostado.
9. Nesta tela de resultado deve ser informar o que o usuário acertou e errou e quanto é o seu lucro, caso exista.

Tema 10: Sistema de controle de ATAS

Objetivo: Trata-se de um sistema da extração e controle das deliberações de uma ATA.

Requisitos:

1. Na tela principal o usuário deve fazer o upload de uma ATA no formato de arquivo de texto.
2. Em seguida deve clicar no botão extrair.
3. A ATA será um documento estruturado em que todos os itens começando pela palavra Deliberação devem ser extraídos.
4. Os itens extraídos formarão uma lista de itens que deve ser exibido na tela. Para cada item da lista o usuário deverá informar a data prevista para o cumprimento da demanda.
5. Em seguida deve cadastrar essas demandas.
6. O programa deve ter no menu a funcionalidade controle de demandas.
7. Na tela de controle de demandas devem ser listadas todas as demandas já extraídas dos arquivos. Se a demanda estiver atrasada, deve ter algum destaque visual.
8. Na tela de controle de demandas o usuário pode editar dando ok para informar o seu cumprimento.
9. As demandas que já receberam o ok, quando listadas, devem ter um destaque visual e a data do cumprimento deve ser exibida.
10. As informações devem ser armazenadas em um arquivo csv.