



INSTITUTO POLITÉCNICO
DO CÁVADO E DO AVE
ESCOLA SUPERIOR DE TECNOLOGIA

RELATÓRIO DE TRABALHO PRÁTICO

SISTEMA DE GESTÃO DE INFETADOS

CARLOS COSTA, DIOGO ALEXANDRE

ALUNO Nº <18830>

ALUNO Nº <18844>

Trabalho realizado sob a orientação de:

Luís Ferreira

Linguagens de Programação II

Licenciatura em Engenharia de Sistemas Informáticos

Barcelos, Maio de 2020

Conteúdo

Resumo.....	3
Introdução	4
Contextualização	4
Motivação e Objetivos.....	4
Estrutura do Documento.....	4
Estrutura de Dados.....	5
Arquitetura	5
Funções Implementadas	6
LoadAll () & SaveAll ()	6
AddPatient ()	6
PrintList()	6
InfectedCases(), CasesByRegion(), CasesByGender(), CasesByAge().....	6
RemovePatientTemp() & ChangeStatus()	6
Conclusão	7
Apreciação Final	8
Bibliografia	9
Anexos	10

Resumo

Neste projeto pretende-se que seja desenvolvida uma solução em C# para um problema real de complexidade moderada, no caso uma ferramenta para auxiliar o sistema nacional de saúde no registo e na contabilização de novos casos infetados numa situação de crise de saúde pública.

Para tal foi necessário a criação de classes ambas com as suas propriedades, construtores, atributos e métodos bem como a implementação de funções para inserção, pesquisa e verificação de pacientes.

Foram também utilizadas funções para contabilizar o número total de infetados, a consulta de casos por região, sexo e idade onde é apresentada a ficha dos pacientes e o estado em que se encontram.

Introdução

Contextualização

A realização deste relatório serve para mostrar e sintetizar todos os parâmetros e documentos utilizados na realização do trabalho prático que foi proposto, para uma melhor avaliação de tudo aquilo que foi utilizado e aplicado.

Motivação e Objetivos

O desenvolvimento deste trabalho tinha em base a criação de um sistema que permitisse gerir pessoas infetadas numa situação de crise de saúde pública, podendo então assim contabilizar todos os casos de pacientes infetados e consultar toda a sua ficha de dados.

Estrutura do Documento

O presente documento encontra-se estruturado por vários parâmetros como se pode verificar pelo índice apresentado anteriormente. Em primeiro lugar, um breve resumo sobre o tema de trabalho bem como algumas palavras-chave e siglas que o grupo achou pertinente. De seguida é feita uma introdução sobre o trabalho realizado, depois uma fundamentação de tudo aquilo que foi aplicado na realização do projeto e por fim uma conclusão e uma apreciação final bem como uma bibliografia.

Estrutura de Dados

Ao contrário da primeira fase, os dados agora serão lidos e guardados num ficheiro criado através da aplicação.

De maneira a processar e ter acesso aos dados do ficheiro, os mesmos foram guardados numa lista para ser possível realizar alterações tais como adicionar, editar e remover um paciente ou alterar alguma informação dentro dos atributos do mesmo.

Quando o programa recebe a ordem de fechar os dados existentes na lista são guardados no mesmo ficheiro de forma a não perder as alterações efetuadas pelo utilizador.

NOTA: Nesta fase só é possível registar um novo paciente ficando este armazenado no ficheiro, quando alteramos o estado do paciente ou removemos um paciente as modificações não são guardadas no ficheiro, contudo são guardadas na lista até o programa fechar.

Arquitetura

A arquitetura implementada no projeto foi o *NTier* de forma a ter cinco camadas:

- Business Objects
- Business Rules
- Data
- Exceptions
- Manage Patients

Funções Implementadas

Foram usadas as seguintes funções no projeto:

- LoadAll
- SaveAll
- AddPatient
- PrintList
- InfectedCases
- CasesByRegion
- CasesByGender
- CasesByAge
- RemovePatientTemp
- ChangeStatus

LoadAll () & SaveAll ()

A primeira função foi implementada para ler/carregar os dados do ficheiro *Data.txt*.

A segunda função foi implementada para guardar um novo paciente inserido pelo utilizador no mesmo ficheiro.

AddPatient ()

Função implementada para criar um paciente na lista onde serão armazenados todas as variáveis referentes aos pacientes.

PrintList()

Função implementada para mostrar todos os dados armazenados na lista.

InfectedCases(), CasesByRegion(), CasesByGender(), CasesByAge()

Funções implementadas para mostrar/consultar dados previamente filtrados por um parâmetro fornecido pelo utilizador. Estas funções também são capazes de contabilizar os casos totais de infetados com o parâmetro em comum.

RemovePatientTemp() & ChangeStatus()

Funções temporárias para editar/remover temporariamente atributos ou um paciente dentro da lista implementada.

Conclusão

A realização da segunda fase deste trabalho prático permitiu desenvolver a capacidade de autoaprendizagem dos elementos do grupo, realizando pesquisa individual sobre os temas lecionados até a data, bem como fazer uma revisão geral sobre toda a matéria lecionada, levando a que todos os conteúdos aprendidos fossem cimentados.

Apreciação Final

Resumindo e concluindo, a realização da segunda fase deste projeto foi interessante e exigiu bastante atenção e cuidado da parte dos elementos do grupo.

Contudo foi crucial para rever toda a matéria lecionada nas respectivas aulas, levando a que o grupo tenha uma melhor performance na próxima fase do trabalho a ser realizado.

Bibliografia

- Ferreira, Luís - 2007-2017, Março 2015 - C# Essencial - EST-IPCA
- Mayo, Joe – 2015 - C# Succintly – USA

Anexos

Em anexo será disponibilizada uma hiperligação para o repositório público do projeto desenvolvido.

[LP2 - TP1](#)