1506
UNIVERSITÀ
DEGLI STUDI
DI URBINO
CARLO BO

CORSO DI LAUREA IN
INFORMATICA APPLICATA
SCUOLA DI
SCIENZE TECNOLOGIE E FILOSOFIA DELL'INFORMAZIONE

Corso di Programmazione e Modellazione ad Oggetti

A.A. 2019/2020

Docente:
Saverio Delpriori
Studente:
Matteo Serafini – Mat. 293168

Specifica

Il progetto consiste nello sviluppare un software che scarichi i dati relativi ai contagi, nazionali e regione per regione, dalla repository della protezione civile, ordinando le regioni per statistiche:

- 1.Variazione Totale Positivi
- 2.Totale Casi
- 3.Guariti
- 4.Positivi
- 5.Deceduti

Il programma dovrà mostrare i dati nazionali a fianco ad una delle classifiche regionali, si potrà poi iterare le altre statistiche tramite eventi.
In particolare ci saranno 3 azioni:

- aggiornare I dati
- selezionare la classifica precedente
- selezionare la classifica successiva

Inoltre mostrerà l’ora attuale così come l’orario abituale di aggiornamento dei dati da parte della protezione civile, e qualche informazione su comandi ed esiti degli eventi.

Studio del Problema

1. Tutte e tre le azioni saranno sempre selezionabili ad ogni stato del programma, tuttavia potrebbero esserci errori legati alla mancanza di connessione o alla mancanza di files da mostrare.

   _Gli errori saranno comunicati tramite interfaccia grafica, per fare questo verranno semplicemente gestite delle Exceptions

2. In visione di una crescita(malaugurata) dei dati si dovrà prevenire il Download dei files non aggiornati, ed in visione di una susseguente implementazione grafica con immagini (tipo grafici a barre…) si dovrà prevenire il caricamento di files già presenti ed uguali nell' interfaccia grafica.

   _Verranno implementati due Proxy Patterns per eseguire gli opportuni controlli ed eventualmente scaricare/caricare i files, i quali, (sia grezzi che ordinati) verranno salvati su file per un successivo riuso.

3. Il programma sarà diviso in stati dove da ogni stato si potrà passare al precedente, al successivo o allo stesso(aggiornando).

   _Saranno creati 3 metodi(uno per ogni scelta) che dovranno essere ridefiniti in base allo stato attuale del programma, per questo si è pensato allo State Pattern

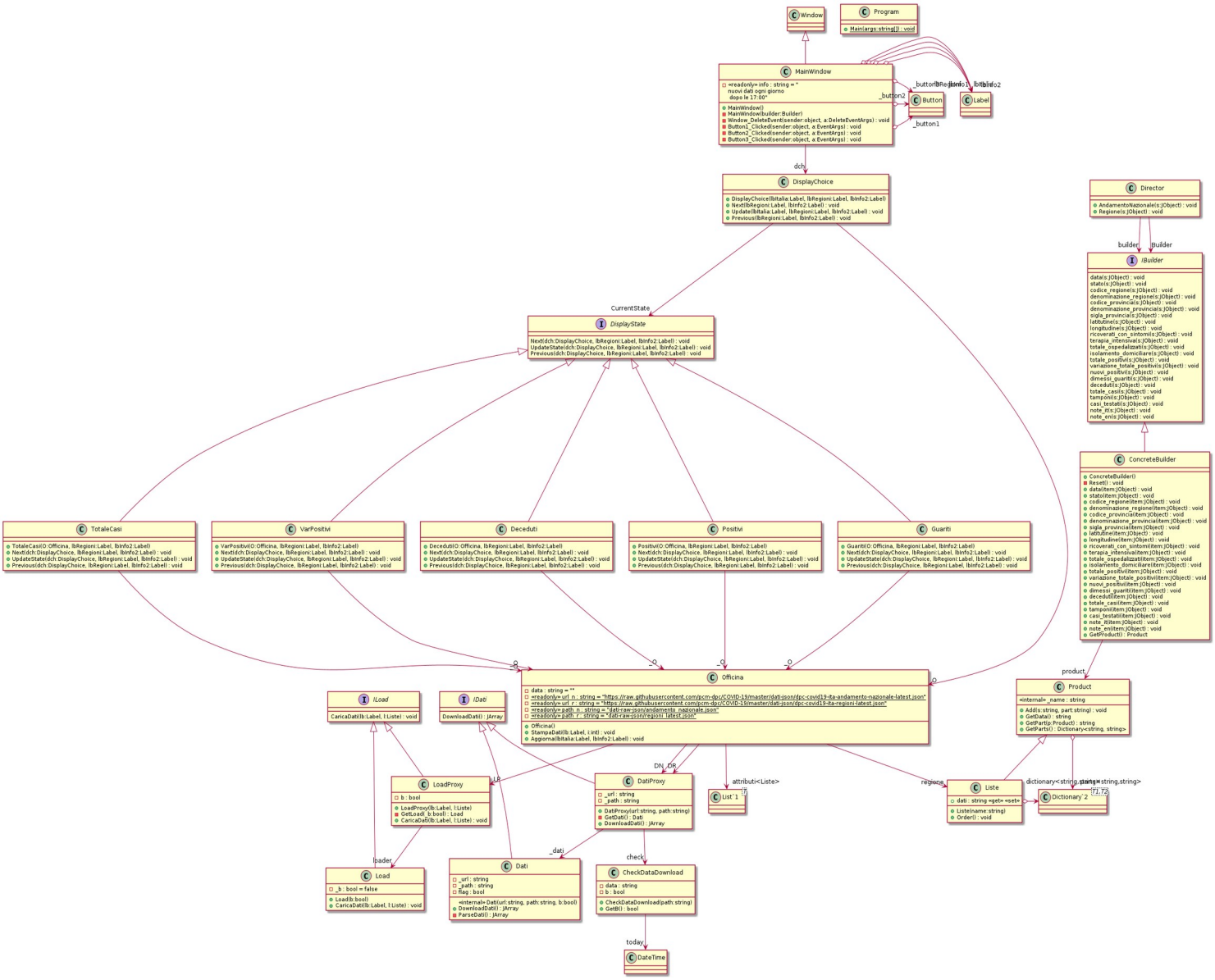
4. Dai dati scaricati, che saranno in formato Json e che dovranno quindi essere decodificati, verrano selezionati degli attributi a seconda del tipo di classifica o lista che si andrà a costruire.

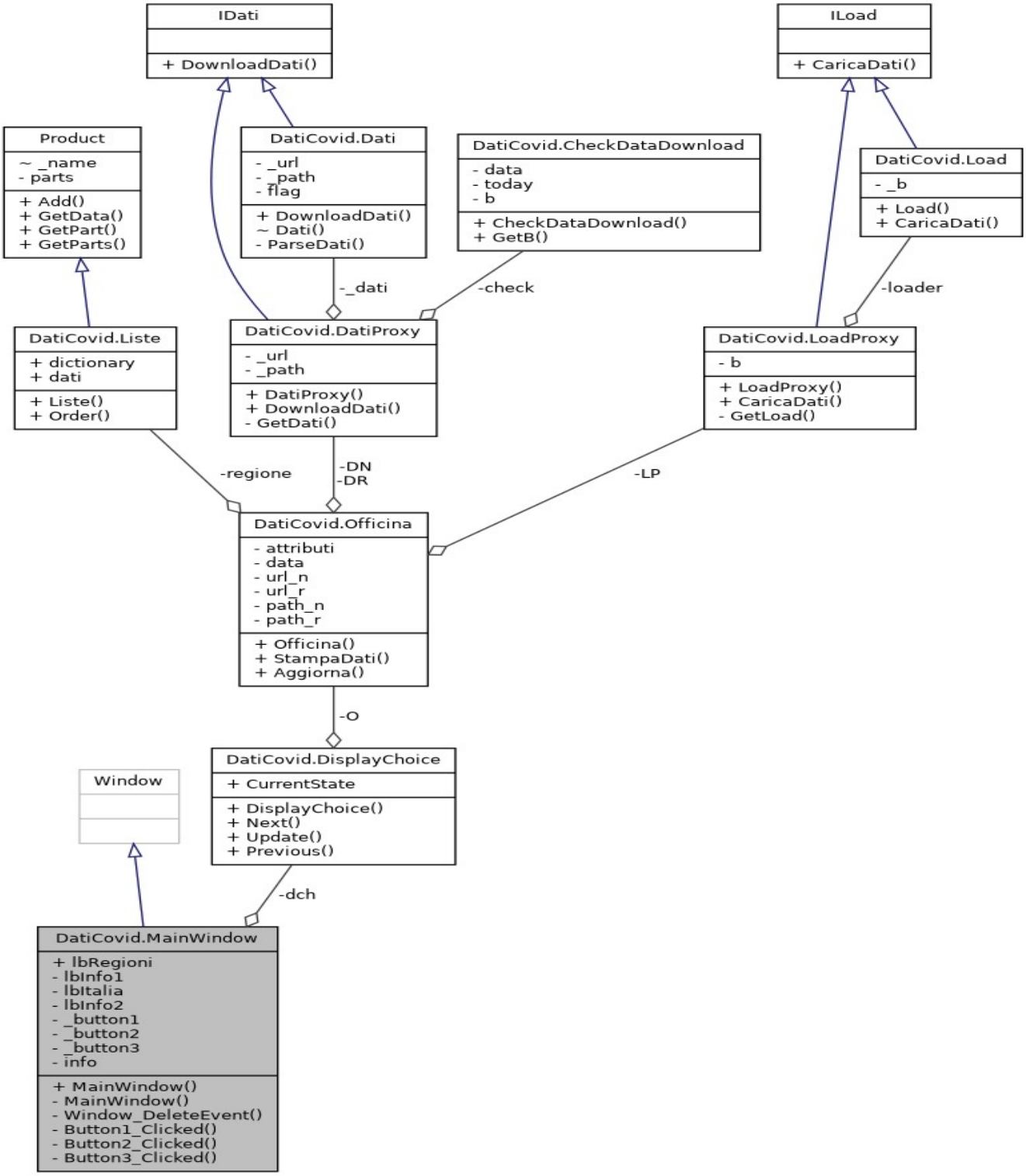
   _Si farà uso di un Pattern Creazionale per creare i diversi tipi di dati,
   e della libreria "Newtonsoft.Json.Linq" per gestire I dati Json.

5. Per la rappresentazione grafica in ambiente linux verrà utilizzata la libreria "Gtk".

Scelte Architetturali

Il programma Main si occuperà esclusivamente di istanziare l' interfaccia grafica, avverrà un primo aggiornamento e/o caricamento dei dati e dello stato del programma, poi il resto dell'esecuzione sarà controllato dall'utente tramite eventi(tasti).
L'interfaccia grafica agisce solamente sulla classe di controllo dello State Pattern che poi si interfaccerà con "Officina", quest'ultima sarà il client di tutti gli altri Patterns(2 Proxy, 1 Builder) I quali si interfacceranno solo con questa(Officina).
I due Proxy Patterns saranno piuttosto standard (interfaccia, classe proxy, classe effettiva) ad eccezione di una classe in più per verificare la "freschezza" dei dati presenti nella macchina.
Il builder pattern dovrà preoccuparsi di costruire 2 tipi oggetti (nazione, regioni) e comprendera una sottoclasse della classe prodotto per definire ulteriori metodi.





Per l’intera documentazione ed I file delle immagini visitare il repository pubblico: “https://github.com/Matteoserafini-lab/PMO_Proj-DatiCovid”

Use Case

Gli Use Case sono solo tre, uno per ogni tipo di evento, dopo la scelta dell' utente ci sarà un aggiornamento di stato del programma ed una susseguente scelta con le stesse opzioni della precedente.

In Particolare:

Use Case: Update

Actor: Utente

Course of events: _l'utente preme il tasto Update
_tramite la classe di controllo dello State Pattern l'Officina aggiorna i dati e stampa la lista nazionale.
_una nuova istanza dello stesso stato viene creata, l'Officina ne stampa i relativi dati aggiornati ed alcune informazioni di contorno.

Use Case: Next / Previous

Actor: Utente

Course of events: _l'utente preme il tasto Next / Previous
_tramite la classe di controllo dello State Pattern viene scelto il successivo/precedente stato.
_una nuova istanza dello stato selezionato viene creata, l'Officina ne stampa i relativi dati (non aggiornati) ed alcune informazioni di contorno.