PROJECT SPECIFICATION

Object-oriented programming - lab in .NET environment

Points:

Learning outcomes					TOTAL
I1	**I2**	**I3**	**I4**	**I5**	
20	20	20	20	20	100

INSTRUCTIONS:

- The defense of the project takes place during the examination periods
- The student applies as well as for written exams
- I1, I2 and I3 refer to Windows Forms technology, and I4 and I5 to Windows Presentation Foundation technology
- 3 days before the defense of the project, the archived project must be sent to the e-mail address of the assistant (eg if the defense is on July 10, the project should be sent no later than July 7, 23:59)
- The project can be done using GitHub. In this case, the student sends a link to his GitHub to the assistant's e-mail address, which must be public, and it is not necessary to send the archived project via e-mail address. As in the case of an archived project, the link to GitHub must be sent 3 days before the project defense.
- Students who defend their work on the deadlines in June and July can receive a maximum grade of 5
- Students who defend the work on the deadlines in September can get a maximum grade of 4 (exception: if the solution fully follows the specification, it is possible to get a grade of 5)
- Students who defend their work on the deadline in January and February can get a maximum grade of 3 (exception: if the solution fully follows the specification, it is possible to get a grade of 4; if good practices in the development of the solution architecture are also applied, it is possible to get a grade of 5)

Introduction

The idea is to create three projects responsible for displaying statistics from the World Cup for men in 2018 and for women in 2019 using the following API: https://worldcup-vua.nullbit.hr (note: if https links do not work, use http protocol).

The projects are as follows: a Windows Forms application, a Windows Presentation Foundation application and a project in charge of data manipulation "Data Layer" (Class Library) which is used in both applications.

Data Layer (Beyond Learning Outcomes)

It is used by both client applications and aims to have all data manipulation done in the data layer.

He is in charge of:

- retrieving data from the specified API (including reading JSON files)
- parsing and mapping of retrieved data
- preparation of data for use in client applications
- data storage in text files
- reading data from text files

API Endpoints explained:

- https://worldcup-vua.nullbit.hr/men/teams/results
- https://worldcup-vua.nullbit.hr/women/teams/results
  - The first endpoint refers to the Men's World Championship, while the second refers to the Women's Championship
  - They contain basic information about each national team (FIFA code, number of wins/losses, ...)
  - Contains the FIFA code required to retrieve details about a particular national team

- https://worldcup-vua.nullbit.hr/men/matches
- https://worldcup-vua.nullbit.hr/women/matches
  - The specified endpoints are used to retrieve detailed information about all matches of the men's and women's championships

- https://worldcup-vua.nullbit.hr/men/matches/country?fifa_code=ENG
- https://worldcup-vua.nullbit.hr/women/matches/country?fifa_code=ENG
  - Examples of retrieving filtered detailed information about England men's and women's national team matches

It is also possible to retrieve data by reading JSON files (downloaded from API endpoints) that you can download from Infoeduke (Materials -> Other course documents -> worldcup-sfg-io.zip). The choice of loading data (API or JSON files) needs to be defined in the configuration file (use the file as desired). When retrieving data, the data layer checks which mode is selected and based on that, the user gets the data.

API review recommendations:

- Mozilla Firefox – recommended because it displays structured data without configuration
- Google Chrome – it is recommended to install the JSONView extension https://chrome.google.com/webstore/detail/jsonview/chklaanhfefbnpoihckbnefhakgolnmc
- Postman – one of the best REST API client

Recommendations for retrieving, parsing and mapping data from client applications:

- https://gist.github.com/acamino/51ae7fa45708bc1e8bcda5657374aa48
- https://code-maze.com/different-ways-consume-restful-api-csharp/
- https://quicktype.io/csharp/

Notes for client projects

It is mandatory to handle all possible exception occurrences correctly. Apps must not crash after launch. It is enough to print a message (dialog box or something similar) that an error has occurred (eg an API call returns an error).

Paths to used resources (images, text files, ...) must not be hard-coded. Relative paths should be used. It is important that the application works normally on any computer that has the appropriate .NET installed without first correcting the program code.

Windows Forms application (Outcomes: **I1**, **I2**, **I3**; points: **60**)

In the Windows Forms application, the functionalities and how they are performed will primarily be evaluated. It is essential that the interface is logical to use and that controls are used that correspond to the requested requirements. The application does not need to be responsive.

Retrieving data from the API must be executed asynchronously, which means that the main thread should only be responsible for plotting data and simple data manipulation. If an operation is not being executed immediately, i.e. it is necessary to wait for the result, it is necessary to display the loading animation.

Initial settings (Outcome: **I1** – minimum, **I2** – desired)

When starting the application, a form will appear in which you need to set the desired priority (female or male) and the language of the application (localization and globalization) - possible languages are Croatian and English. The selection must be saved in a text file on disk and used every time the application is started. If the file does not exist, the user should be asked again to select the priority and language.

Note: resources fetched from the API do not need to be localized (e.g., country name).

Favorite national team (Outcome: **I1** – minimum, **I2** – desired)

When the application opens, it is necessary to select the favorite representation from the ComboBox control. The national teams must be retrieved from the previously processed API and printed in the form "NAME (FIFA_CODE)". The selection must be saved in a text file and used during subsequent loading of the application.

Favorite players (Outcome: **I2** – minimum, **I3** – minimalni and desired)

After choosing your favorite national team, you need to choose your three favorite players. All players can be retrieved from the data on the first played match as a union of the "starting_eleven" and "substitutes" sets.

Players will be displayed in a user-defined control that contains name, number ("shirt_number"), position, information on whether he is the captain and whether he is a favorite player (show a star next to the favorite player).

Favorite players should be displayed in one Panel control, and other players in another Panel control. Switching the player from one panel to another can be done by selecting from the context menu and dragging (Drag and drop) the user control to another panel. Also, it is possible to mark several players at once and transfer them to another panel. The selections need to be saved in a text file and used during subsequent loading of the application.

Player pictures (Outcome: **I2** – minimum)

It is possible to set a picture for each player from the selected national team. Images need to be stored within the solution itself (same as files) so that they can be used in both client projects. The image should be part of the player user control. If a player does not have an image set, a default image of the player should be displayed.

Ranking list (Outcome: **I1** – desired)

Create player rankings based on the following criteria: number of goals scored and number of yellow cards - it is necessary to print the player's full name and display the player's picture along with the number of appearances.

Create a ranking list of the number of visitors for a specific match - it is necessary to display the location, number of visitors, the name of the host ("home_team") and the name of the guest ("away_team").

Note: all rankings refer to the selected national team.

Print ranking list (Outcome: **I1** – desired)

Previously processed rankings must be able to be printed using the selected printer (it is sufficient to create a PDF document). All information (all players and all matches) should be visible on the printout itself, as well as on the tabular display in the application.

Closing the application and setting (Outcome: **I1** – minimum and desired, **I3** – minimum)

It is possible to change the selected championship and application language through the "Settings" form. It is necessary to request confirmation from the user to set priority and language. The "Confirm" option can be executed with the help of the "Enter" key on the keyboard, and the "Cancel" option with the "Esc" key on the keyboard. The confirmed selection must be saved in a text file.

When closing the application, it is necessary to ask the user for confirmation if he really wants to do this. Confirmation can be done using the keys on the keyboard in the same way as already described for changing the settings.

<u>Windows Presentation Foundation (WPF) application</u> (Outcomes: **I4**, **I5**; points: **40**)

In the Windows Presentation Foundation application, apart from the functionality itself, it is important that the application be responsive. The interface should be logical to use and the controls should correspond to the required requirements. Retrieving data from the API must be done asynchronously. If an operation is not executed immediately, i.e. it is necessary to wait for the result, it is necessary to display the loading animation.
The priority and language of the Windows Forms application will be used in the WPF application as well. It will be possible to change the title and language through the WPF application, so the same will apply to the Windows Forms application.

Initial settings (Outcome: **I4** – minimum, **I5** – desired)
When starting the application, a window will appear in which it is necessary to determine the basic settings of the application, namely the desired preference and language of the application (if not already set), as well as the display mode of the main window: fullscreen or selected window size (offer at least 3 window resolutions). In the case of any display mode, the application must always be responsive.
The selections must be saved in a text file on the disk and used each time the application is started. If the file does not exist, the user should be asked again to configure the settings.

Overview of the national team (Outcome: **I4** – minimum, **I5** – minimum)
When loading the application, it is necessary to load the preferred representation set in the Windows Forms application. It is possible to change it through the ComboBox control. In addition, it is necessary to select one of the national teams that played against the selected national team. National teams should be displayed in the form "NAME (FIFA_CODE)". After selecting both national teams, the result of the match played between the selected national teams should be visible (e.g., "2 : 1").
Next to the selected national teams, buttons should be placed which, with an animation lasting 0.5 seconds, will open a new window in which basic information about the selected national team is visible: name, FIFA code, number of games/wins/losses/draws, goals scored/conceived/ difference.

Display of the initial setup (Outcome: **I4** – desired, **I5** – desired)
Below the previously selected national teams, it is necessary to draw a representation of their starting lineups ("starting_eleven"). The above-mentioned display should have a rectangular representation of the field with a clearly marked middle of the field (it is sufficient to place a background image of the football field). Players should be placed on each half of the field in the form of user-defined controls. The positions of the placed players must correspond to their positions in the selected match. Possible positions are: goalkeeper ("Goalie"), defender ("Defender"), midfielder ("Midfield") and forward ("Forward"), and can be read from the "position" attribute in the "starting_eleven" collection. Examples: https://postimg.cc/BPQH5VDs, https://postimg.cc/5j7CdpLY, https://postimg.cc/jDz25hwh.
The player user control must contain: name, number ("shirt_number") and image placed in the Windows Forms application.

Player overview (Outcome: **I4** – desired, **I5** – minimum)
By selecting one of the players from the starting line-up display, with an animation lasting 0.3 seconds (must be different from the animation used for the national team preview), a new window opens containing information about the player: name, number ("shirt_number"), picture, position , information on whether he is the captain, the number of goals scored (in that match) and the number of yellow cards (in that match).

Closing the application and settings (Outcome: **I4** – minimum)
The set application settings can also be changed through the "Settings" window. It is necessary to request confirmation from the user for changes. The "Confirm" option can be executed with the help of the "Enter" key on the keyboard, and the "Cancel" option with the "Esc" key on the keyboard. The confirmed selection must be saved in a text file.
When closing the application, it is necessary to ask the user for confirmation if he really wants to do this. Confirmation can be done using the keys on the keyboard in the same way as already described for changing the settings.