OOP-Project
Team Topaz – Organizer
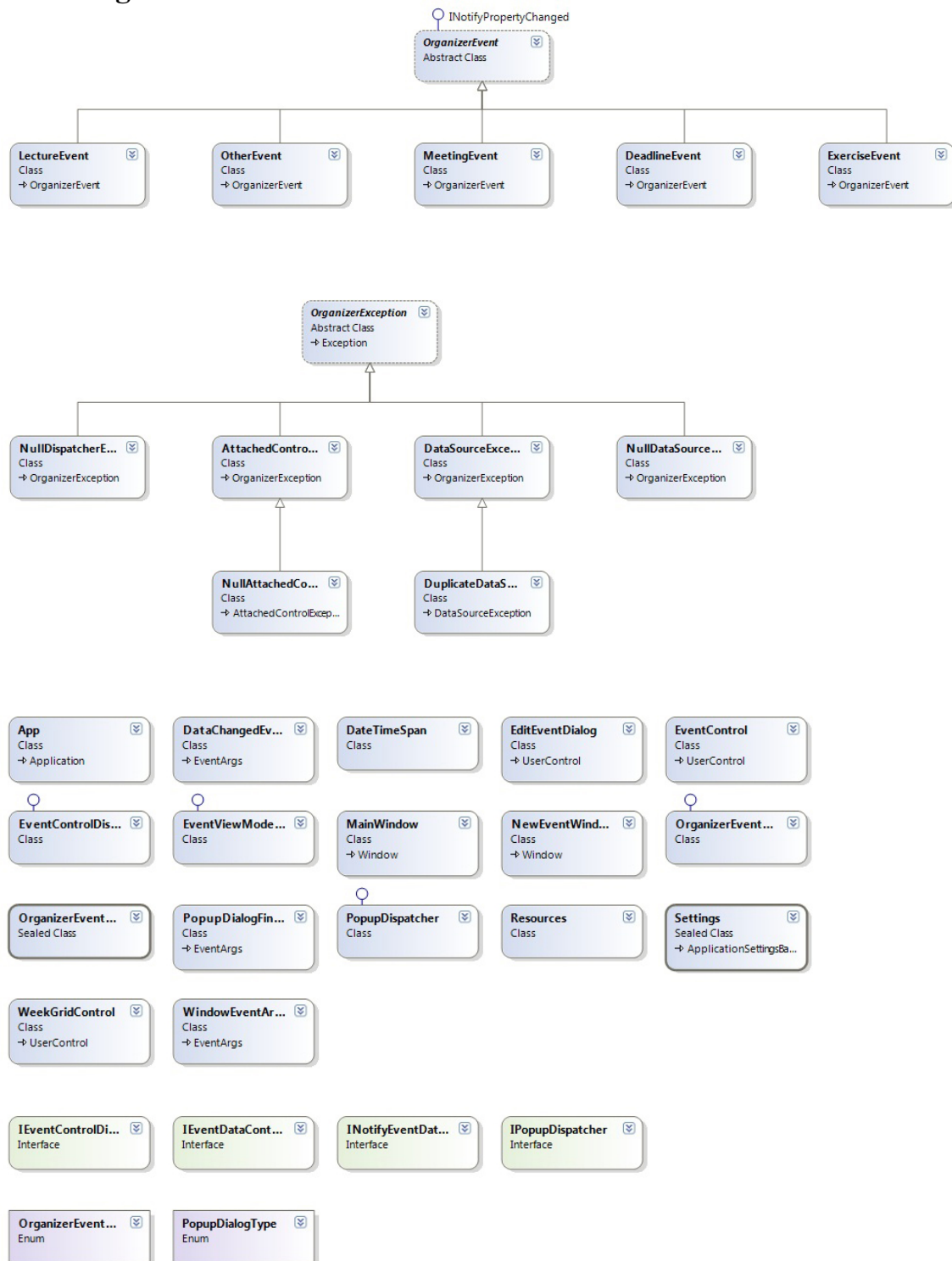
Team name: Topaz

Team members:
- Penka Kamenova *(username: pepakam)*
- Georgi Neykov *(username: georgi.neykov)*
- Ivan Zdravkov *(username: IvanZdravkovBG)*
- Kolio Kolev (*username: Quasar)*
- Gabriela Angelova *(username: kalbo_17)*

Project purpose:
Our project represents our notion of a good organizer. The following application provides options like adding events with start time and end time to the timeline, editing already existing events, adding and removing notes without time limit and saving the whole information in an improvised database (.txt files). Our program has the ability to load all the existing events and notes every single time we run it. You can also see the current date and time and the remaining part of the day with only one look at the organizer grid. As a deeper description of our project we append Scheme 1.

EDIT: By opening the program, you are asked to enter a username and a password. Both of them are *123*.

Class diagram:

SVN repository:

Scheme 1:

Topaz Organizer Application

Model

ExamEvent

ProjectEvent

ExerciseEvent

LectureEvent

MeetingEvent

JobEvent

DeadlineEvent

OtherEvent

SchoolEvent
- Base class for all school related events
- Contains information about course name, lecturer, room

OrganizerEvent
- Base class for other event classes
- The basic container of information for an event's details

DataWriterReader
- Saves/reads the events data holder contents.

IEventDataImporterExporter

OrganizerEventData
- Holds a collection of OrganizerEvent objects
- Notifies subscribers about changes in it's contents.

Global

App class
- Main class of apllication

GlobalExceptionHandler
- Catches all possible exceptions.
- Creates a MessageBox in the app window showing information for the error.
- Acts different on fatal and non-fatal exceptions.

Some other exceptions

OrganizerEventParameterException
-To be thrown on invalid input parameters relating the contents of OrganizerEvent object
- Non-fatal exception, program continues to run.

OrganizerException
- Base class for all custom app exception
- Should inherit ApplicationException class from. NET

Controller

EventViewModelDispatcher
- Modify contents of WeekGridControl based on the data model contents by using EventControls.
- Handle user input from popup controls.
- Open, close, position Popup controls.
- Create, remove, size & position EventControls on WeekGridControl
- Add, edit or remove contents from the data model.
- Take action on modification of the contents of the data model.

PopupDispatcher
- Open, close, position Popup controls.
- Check user input
- Communicate with the main dispatcher.

EventControlDispatcher
- Create, remove, size & position EventControls on WeekGridControl
- Communicate with the main controller.

ModelView

MainWindow

WeekGridControl
- The container for all EventControls
- Shows the days and hours of a single week in a grid
- Gets mofied and filled by the Controller

Other WPF Controls

EventControl
- Represents a OrganizerEvent on the weekgrid
- On creation binds itself to the datasource's properties
- Handling and positioning is performed by the controller

CreatePopupControl
- Opens when an empty cell from the grid is clicked.
- Collect input from the user and sends a new OrganizerEvent to the Controller, which then adds it to the Model (no direct calls to Model).

EditPopupControl
- Opens when an existing EventContol is clicked
- Binds to the EventControl that called it and edits its data.
- Should not use any direct calls to the objects from Model.