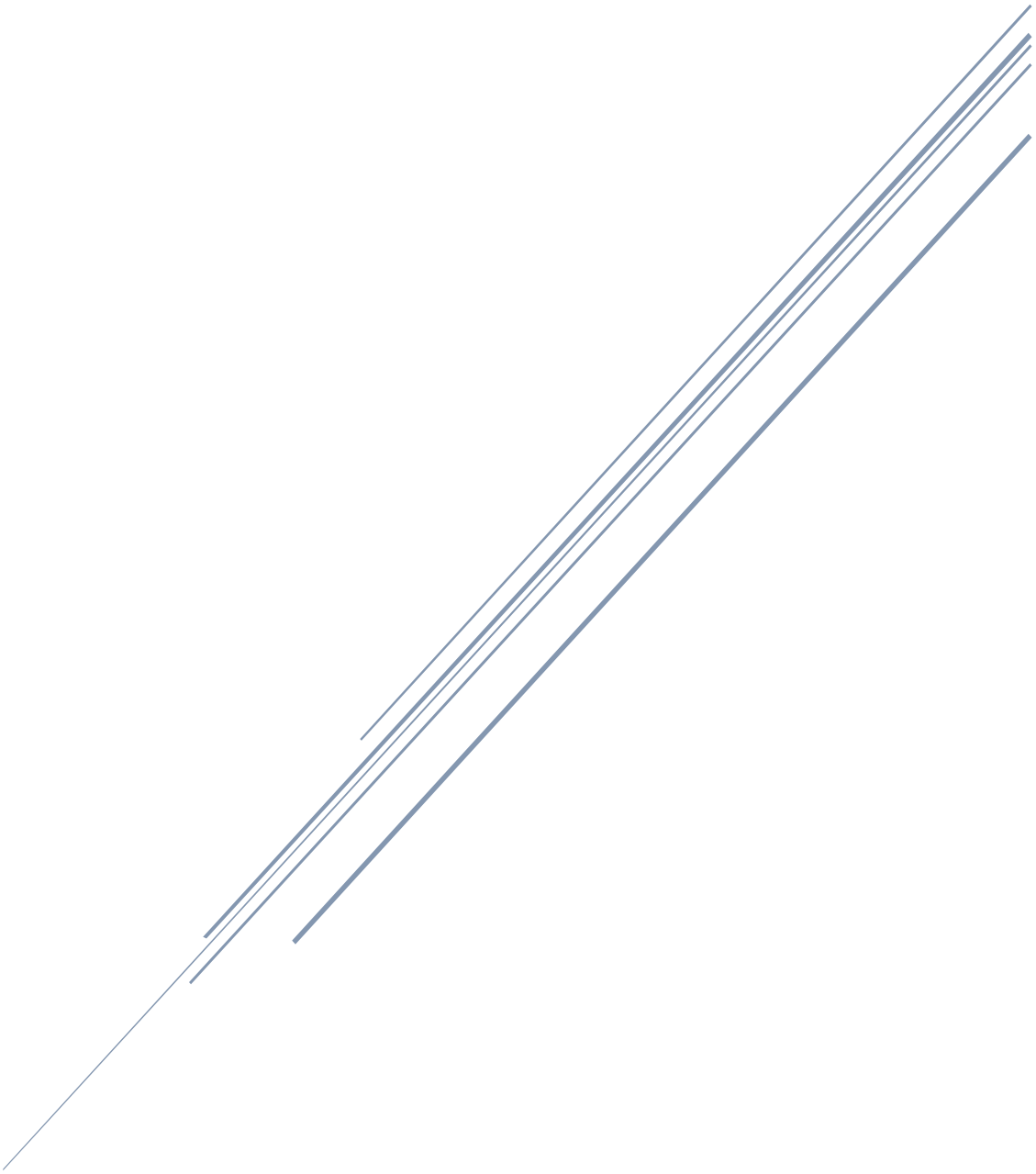
YOUR GRADEBOOK

Documentation project coding phase



TBZ

Modul 326

Content

Document description

In this documentation we have noted down the most important parts of our project which is called “Your Gradebook”. What does this project do? Well for that you will see a project description in the documentation. It also contains a small timetable which gives an idea of all our steps. With this documentation a person can get a better idea what we have been doing in the coding phase of this project.

Project descrption

Our idea is to program an application called “Your Gradebook”. In this application you can keep an overview of all your marks. If you go to two different schools (BMS/ABU) then you also have the option to work with more than one gradebook. But you can’t look into two gradebooks at once. At the starting it asks for your name and opens an account. Then a small menu pops up which asks you if you want to add a mark or look at the overview. If you choose to add a mark then it first asks for the school you go to. After that is asks you which subject you want to add in your gradebook. In the gradebook you can add your subjects and name them however you want. There is no fixed number of subjects so you can add as many as you please. After choosing a subject you can add a mark and also give in an evaluation for each mark, that means how much that mark weighs for your average. If you choose the option “overview” in the menu then it will show you all your subject and the average of your subjects this semester. If the same user logs in for the second time, all of his/her previous data will be saved. Such as school or subject.

Tools

Design phase:

- Description and Documentation: Office Word version 2019
- Use Cases, Class diagram, Domain Model: draw.io ()

Coding phase

- Timetable: Office Excel Version 2019
- Documentation: Office Word Version 2019
- Coding: Visual Studio 2019

Time table

We made a timetable to make sure we are on the right track during the whole project. At least in the implementation phase. We first made a checklist of everything we needed and then started distributing them in lessons and dates. As we had already made sure that we finished all the diagrams and charts in the design phase, we already knew what had to be done. We also distributed the tasks so we exactly know who need to do what. Once we distributed everything and the timetable made sense we were ready to start with our project. Then obviously our next steps were to implement the grade book we needed and last but not least make sure testing is done. We didn't mention the documentation in the timetable as we made sure we will write the documentation continuously.

Person who does it
Tamanna Rajput
Anushya Nagulanandan
Both

Date	***Task***	***Person***
15.03.2021	Make a time table	
22.03.2021	Implement Class Starter	
	Implement UserInput	
	Implement Calculator	
29.03.2021	Implement Serializer	
	Implement Subject	
05.04.2021	Implement Marks	
	Implement SerializableMarks	
12.04.2021	Do Testing	
	Finish Documentation	
19.04.2021	Check everything before submission	
	Submission	

Work Process

Of course, the whole work process was based on the timetable. We first started creating the base and the structure of the project together to make sure it's the same as the class and domain diagram. The whole design phase definitely helped us as the implementing went on faster. We also created the object and methods together. Just the logic in the induvial methods had to be implemented. We had to slightly change something which made our simplified our code. We added a new class called school and made sure we used the property in the class subject. Then we both split up and worked on the induvial classes and started implementing the logic. We still did a bit of pair programming because the GitHub with two different IDE's wasn't working but still managed to do a team work. We started with the slightly harder classes which were UserInput, Calculator and Serializer. These all had different logic and objects which took us time to implement. After these classes we went on with the classes Subject, Marks, School and ObjectToSerialize. These classes were pretty similar. They all had the method GetObjectData which basically got the needed object for saving or getting something. Then the classes Marks and ObjectToSerialize also had the method ToString which converted the object into a string. Once we were done we made sure that all the classes are integrating with each other as it was also planned. We went through a few trial rounds and implemented few test data so we could use it in the future for testing purpose. When the testing part came we wrote some test cases which we made sure are very similar to the use cases so that it makes sense. Then we went through every test case and wrote down what was expected and what was the result. Once we were sure that everything was completed, worked well and was ready, we submitted our project successfully. Of course, the whole documentation was not just written at the end of the project as it would be a really big hustle. We made sure that during the project we updated the documentation when need. The only thing we wrote at the end was the reflection of our whole work as a team in module 326.

Testing

We wrote test cases so that we can reassure that all the functions we implemented in our program are working as expected. It helps us validate if the program is free of defects or does something need to be changed. We made sure that our test cases ensure good test coverage. We also tested our validations in which we on purpose told the user to do something which is incorrect. This helps us increase out coverage and test quality. At the end of the program we managed to make the code as limited with problems as we could. In total we had "16" test cases.

Testcase Nr.	Testcase	Expected result	Successfull	Remarks
1	Type in User name	It should give you two options which you can choose from: "Add marks" / "Overview Marks"	x	-
2	Choose Add Marks	It should ask you which school	x	-

3	Choose option add school	It should ask you for a school name	X	-
4	Give in a school name	It should ask you to create a subject	X	-
5	Choose the option to create a new subject	It should ask you for the subject name	X	-
6	Type in a subject name	It should ask you to give inthe name/title of the mark/test	X	-
7	Type in the name/title	It should ask you for the mark	x	-
8	Type in a mark	It should ask you to give in th weight of the mark	x	-
9	Type in the required weight	It should ask you weather you want to continue or not	x	-
10	Type in no	The program should end and you should see a message	x	-
11	Go throught test cases 1-9. Type in yes after the 9th test case	It should give you two options which you can choose from: “Add marks” / “Overview Marks”	x	-

12	Go through test cases 1-3	There should be a saved school name from previous option	x	-
13	Go thorugh test cases 5 & 7	There should be a saved subjects name from previous option	x	-
14	Continues with the program so that you land on the expected result of test case 11 again	It should give you two options which you can choose from: “Add marks” / “Overview Marks”	x	-
15	Choose Overview Marks	It shows and the avergage of the student / It should ask you weather you want to continue or not	X	-
16	Go through test cases 10 & 11 for good test coverage	Looked at the expected test result of test case 10 & 11	x	-

Reflection

Overall, we think we did a pretty good job implementing this whole project. Not just the design part but also the coding part. We both are very proud of the result. The most important thing is that we tried doing something in C# for the first time and it worked. We think that that’s a big achievement. At the end we had a small problem where we couldn’t compile our code on Visual Studio which is the best tool for C#. We really panicked but managed to solve the problem on a different IDE and configured the project setting so that it also worked on Visual Studio. The only thing we let slip out of our mind was building a design pattern in our code. At the end of the project once we submitted it, we were more than happy to say that we worked hard and did a good job.