

Bucharest University of Economic Studies

Faculty of Cybernetics, Statistics and Economic Informatics

Economic Informatics Program

Integrated Student Dashboard and Academic Resource Management

Student:

Vlad Liviu-Alexandru

Bucharest 2024

Declaration regarding the originality of the content and the assumption of responsibility

I hereby declare that the results presented in this paper are entirely the result of my own creation, unless references are made to the results of other authors. I confirm that any material used from other sources (magazines, books, articles and Internet sites) is clearly referenced in the work and is indicated in the list of bibliographic references.

Contents

[1. Presentation of the computer system 4](#_Toc155546949)

[1.1 General Description 4](#_Toc155546950)

[1.1.1 The problem to be solved 4](#_Toc155546951)

[1.1.2 The goal 4](#_Toc155546952)

[1.1.3 The context 4](#_Toc155546953)

[1.2 Requirements specification 5](#_Toc155546954)

[1.2.1 General and detailed Use Case diagrams 6](#_Toc155546955)

[1.2.2 Textual description of use cases 7](#_Toc155546956)

[2. Analysis of the system 13](#_Toc155546957)

[2.1 Activity Diagrams 13](#_Toc155546958)

[2.1.2 Activity diagram for uploading and downloading files 13](#_Toc155546959)

[2.2 Class Diagram 13](#_Toc155546960)

[2.3 State Diagrams 13](#_Toc155546961)

[2.4 Interaction Diagrams 13](#_Toc155546962)

[2.5 Processes and collaboration diagrams in BPMN 13](#_Toc155546963)

Figures

Figure 1 - UC01 - General use case diagram 6

Figure 2 - UC02 - File upload 6

Figure 3 - UC03 - Using forum 6

Figure 4 - UC04 - Edit and view progress 7

Figure 5 - AD01 - File upload activity diagram 13

Figure 6 - AD02 – Forum usage activity diagram 14

Figure 7 - AD03 - Progress tracking activity diagram 14

Figure 8 - CD01 - Class Diagram 15

Figure 9 - SD01 - File entry state diagram 16

Figure 10 - SD02 - Forum post state diagram 16

Figure 11 - SQ01 - Add first year students sequence diagram 17

Figure 12 - SQ02 - Generate class repartition sequence diagram 17

Figure 13 - CM01 - Track progress communication diagram 18

Figure 14 - PD01 - Generate calendar view process diagram 18

Figure 15 - CL01 - Forum discussion collaboration diagram 19

Figure 16 - Application detailed class diagram 20

Figure 17 - API detailed class diagram 20

Figure 18 - Database schema 21

Figure 19 - UI - Login 21

Figure 20 - UI - Dashboard 22

Figure 21 - UI - Progress 22

Figure 22- UI - Files 22

Figure 23 - UI - Forum 23

Table 1 - UC02 - File upload use case textual description 8

Table 2 - UC03 - Using forum use case textual description 10

Table 3 - UC04 - Edit and view progress use case textual description 11

1. Presentation of the computer system

1.1 General Description

1.1.1 The problem to be solved

Students are exposed to more knowledge than ever before in the modern educational environment. This abundance of knowledge allows them to surpass yesteryear’s generation within an even smaller time window. However, there are advantages as well as disadvantages. This is also the case for our students, who lose out on knowledge and become less efficient as a result of these resources being dispersed over several websites, platforms, emails, and courses even if it is technically available.

My own observations show that students frequently create groups to tackle this very problem. A few of them are assigned the responsibility of creating a more thorough and user-friendly list or hierarchy of the material that is accessible, since many find it to be not worth the effort.

For every given task, there are a plethora of tools available, which in our case equates to tens of platforms for knowledge and education. Then, organizing and compiling the information in one location becomes the issue rather than having or locating the information. From programming platforms that are enormously helpful to insightful statistics databases, it’s redundant to create another competitor for these websites, as it only makes the problem worse with yet another platform to remember.

1.1.2 The goal

What students need is a platform that allows them to gather all of this information in one place, sort it in a more concise way and enable them to share their finds with ones that come after. We want to enhance their experience and ease their path into new subjects and technologies, not overwhelm them with the gathering of this knowledge. The know-how exists and it’s been documented; Yesteryear’s students faced the same problem and had to deal with the exact same tasks and it quickly becomes clear there is redundant work that could be saved for others to follow.

Thus, the solution describes itself: A student’s dashboard and learning platform that foregoes providing information and instead turns our attention to the gathering and sorting it. Categorizing websites and files based on their respective field, public message boards and more; All meant to place the resources back in our students’ hands.

1.1.3 The context

The application will be used by students regardless of their year of study or experience. It allows them to store private files, homeworks, projects and categories and filter them by the subject and also see publicly shared general interest resources. In addition, comment sections for the posts are also available and a dedicated forum-style questions area for solving common dilemmas for your average student.

1.2 Requirements specification

The Integrated Student Dashboard application is designed as a web application, developed with the help of ASP.NET. It is a stand-alone application that can handle data transfers and storage in order to provide students the possibility of uploading homeworks, projects and general resources for any given subject. A Microsoft SQL Server database and an ASP.NET Core API are used to provide the needed authentication and authorization to ensure confidentiality and safety of all of the student’s data. All data is retrieved trough the API ensuring private data is only available to their rightful possessor. There are three types of accounts: the student’s, the teacher’s which has limited moderator capabilities which include, but are not limited to, creating public posts, sending study group level messages, upload materials, etc. Also, there are the true administrator accounts which have the most leeway in modifying the hierarchy. The administrators are the only ones capable of creating and deleting accounts, assigning groups and classes and much more.

[TO BE CONTINUED]

1.2.1 General and detailed Use Case diagrams

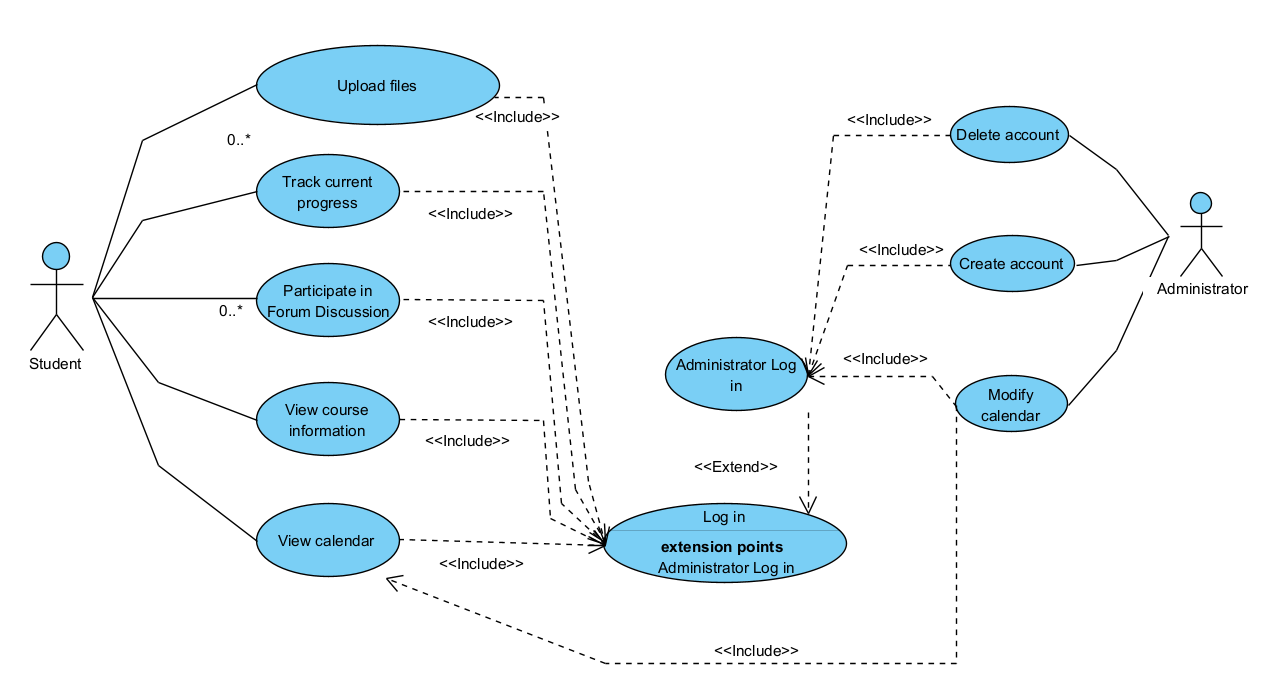


Figure 1 - UC01 - General use case diagram

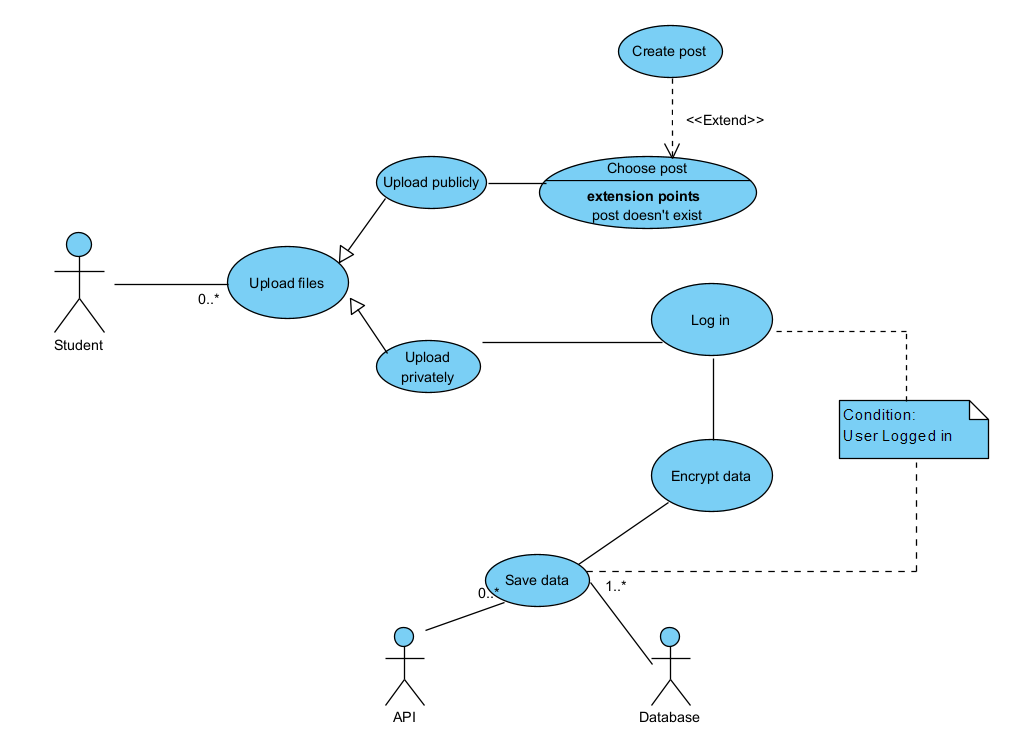


Figure 2 - UC02 - File upload

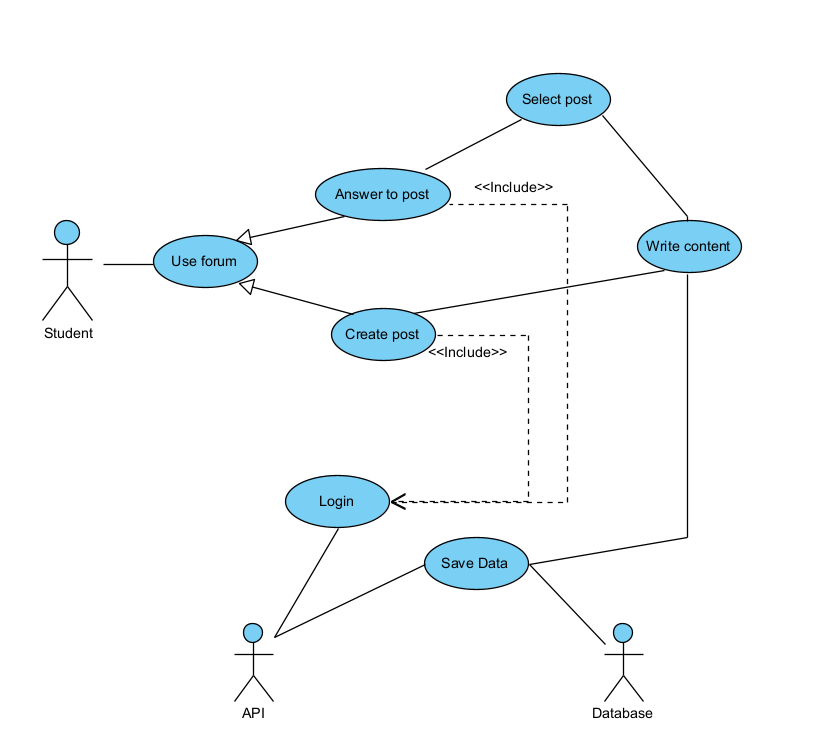


Figure 3 - UC03 - Using forum

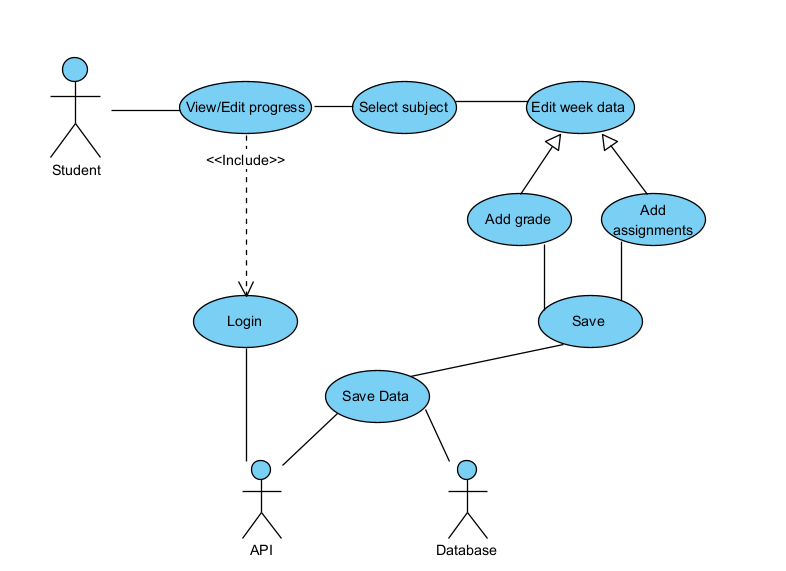


Figure 4 - UC04 - Edit and view progress

1.2.2 Textual description of use cases

1.2.2.1 UC01 – General Use Case Diagram

The diagram presents the 5 most important and frequently used actions the average user might interact with. As the application is tailored to facilitate the general management of a student’s resources and time, these actions represent the main activities a user might want to interact with to achieve the 5 main needs to achieve this goal.

Firstly, resource management is achieved trough the use of the file upload action which, as the name implies, facilitates resource storage by allowing the user to associate more metadata about the resource in question upon upload including, but not limited to: related course, related seminar or week, short description for later search.

Secondly, for the need to gain a bird’s eye view on the current state of the school year and overall classes, there is the track current progress action which in basic terms allows the user to see a timeline of past and future events like tests, exams and projects, whilst encompassing a better understanding of the overall importance and urgency of each one trough the implicit visual comparison of the due dates. Related to this action are also the view calendar and view course information. Although different in nature, all 3 are similar in principle, creating a structured insight of the current active courses whilst fulfilling different needs. The course information shows you a more detailed overview including other information than the one present in other sources like the faculty’s website like uploaded files related to that information and calendar notes. The calendar also shows you an insight, but to the temporal aspect of the study management need, encompassing a more visual and dynamic way of tracking future events, official or otherwise. Allowing meeting planning, homework tracking and more.

The last, but definitely not the least, is the participate in forum discussion which is by far the most complex of the actions, branching into many other sub-actions. This action refers to the forum environment of the platform which serves as the middle-ground between students with a more rounded programming and IT experience and ones looking to learn both course or not related information.

1.2.2.2 UC02 – File upload

The File Upload action stands as a very important component of the application when it comes to resource management, offering users a streamlined method to store and organize their resources. This feature allows users to upload files while providing a robust framework for associating comprehensive metadata with each resource. The user, upon initiating an upload, is presented with the possibility of associating multiple pieces of information about the resource in question to further facilitate the sorting and searching process.

One very notable aspect of the File Upload functionality is its categorization of essential contextual information. Users can link the uploaded resource to specific elements, such as related courses, seminars, or designated weeks within a curriculum. This association not only ensures a structured organization of resources but also facilitates quick and efficient retrieval when needed.

Moreover, the action goes beyond mere categorization by enabling users to provide a succinct yet informative short description for each uploaded resource. This serves as a valuable tool for future search endeavors, allowing users to recall key details about the uploaded file swiftly. Whether it be a brief summary, relevant keywords, or specific attributes, the short description feature enhances the searchability of resources, contributing to a more efficient and user-friendly resource management system.

In essence, the File Upload action is not just a mechanism for transferring files; it is a tool that empowers users to enrich their resources with context and metadata, fostering a comprehensive and organized approach to resource management within the system.

|  |  |
| --- | --- |
| **Element of the Use Case** | **Description** |
| Code | UC02 |
| State | Outline |
| Purpose | Uploading a file or resource |
| Name | File upload |
| Main Actor | Student |
| Description | Description of the file upload mechanism |
| Preconditions | Being logged in |
| Post Conditions | The file is uploaded to the database |
| Trigger | User initiates file upload. |
| Basic Flows | 1. The user opens the files page  2. The user selects the method to upload a new file  3. The user enters the additional information about the resource  4. The user saves the new added resource  5. The file is encrypted  6. The file is saved to the database |
| Alternate Flows | 1. The user opens the files page  2. The user selects the method to upload a new file publicly  3. The user enters the additional information about the resource  4. The user creates a new post  5. The user enters the post title and body  6. The user saves the new added resource  7. The file is encrypted  8. The file is saved to the database  9. The post is created |
| Relationships | N/A |
| Frequency of Use | Frequent |
| Business Rules | The file must satisfy a size and format requirement |

Table 1 - UC02 - File upload use case textual description

1.2.2.2 UC03 – Using forum

The Forum Usage feature serves as a vital component of the application, particularly in the context of collaborative discussions. It provides users with an efficient means to engage discussions, facilitating seamless communication and information exchange. This functionality allows users to participate in discussions, share insights, and contribute to various topics, creating a dynamic and interactive community.

A notable aspect of the Forum Usage functionality is its organization of discussions into relevant categories and topics divided into school or non-related categories and further into relevant and easy to find sub-categories. This categorization not only enhances the overall user experience but also facilitates quick access to information on specific themes.

Furthermore, the feature goes beyond simple categorization by allowing users to add tags and files to their posts, providing additional context. Tags serve as descriptive labels, summarizing the key themes or topics within a post. This tagging system enhances the searchability of discussions, enabling users to find and contribute to conversations that align with their interests more efficiently.

|  |  |
| --- | --- |
| **Element of the Use Case** | **Description** |
| Code | UC03 |
| State | Outline |
| Purpose | Interacting with the forum |
| Name | Using forum |
| Main Actor | Student |
| Description | Description of the forum structure and usage |
| Preconditions | Being logged in |
| Post Conditions | A post is created, modified or deleted |
| Trigger | User enters the forum |
| Basic Flows | 1. The user opens the forum page  2. The user chooses the create post action  3. The user creates the title and body of the post  4. The user optionally chooses other attachments  5. The post is saved to the database |
| Alternate Flows | 1. The user opens the forum page  2. The user chooses a post  3. If the user chooses to delete the post, the delete button is pressed and the post is simply deleted  4. If the user chooses to modify the post, the modify button is pressed, and steps 3 to 5 from the basic flow are resumed |
| Relationships | N/A |
| Frequency of Use | Frequent |
| Business Rules | The post must satisfy size constraints |

Table 2 - UC03 - Using forum use case textual description

1.2.2.2 UC04 – Edit and view progress

The edit and view progress actions are part of the progress management feature, providing a simple and concise way of organizing past and future events related to school activities. Although simple in nature, data added to the progress page goes a long way in creating a more inclusive and seamless experience by automatically processing the data into calendar entries, course information and possible optional relation specifiers for stored files.

|  |  |
| --- | --- |
| **Element of the Use Case** | **Description** |
| Code | UC03 |
| State | Outline |
| Purpose | editing and viewing progress information |
| Name | Edit and view progress |
| Main Actor | Student |
| Description | Description of progress management usage |
| Preconditions | Being logged in |
| Post Conditions | A progress entry is created and/or edited |
| Trigger | User opens the progress page |
| Basic Flows | 1. The user opens the progress page  2. The user chooses a subject  3. The user selects a week or date  4. The user adds information to the entry, either grades or assignments  5. The entry is saved to the database |
| Alternate Flows | None |
| Relationships | Calendar, Forum, Files |
| Frequency of Use | Frequent |
| Business Rules | The post must pass formatting rules |

Table 3 - UC04 - Edit and view progress use case textual description

2. Analysis of the system

2.1 Activity Diagrams

2.1.1 Activity diagram for uploading files

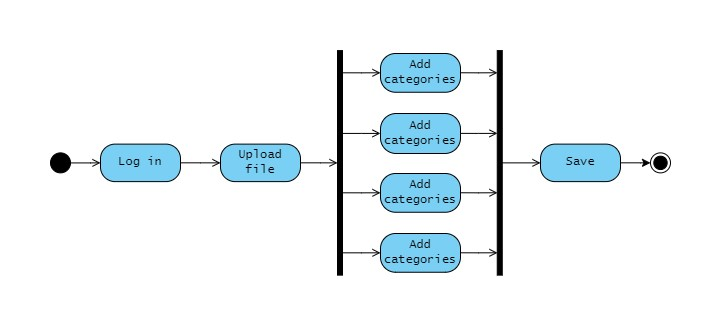


Figure 5 - AD01 - File upload activity diagram

The above-shown diagram exemplifies the workflow for the file upload activity. In this process, users initiate the action by opening the file tab and pressing the “Upload File” button. Upon choosing this option, users are prompted to upload a file. Before continuing to the next step the file’s size and format are verified. Afterwards, the user enters metadata related to the resource, such as the associated course, seminar or week, and a brief description for effective search and categorization.

Simultaneously, the file is prepared for storage using loss-less compression and encryption while the user fills in the information and is then send to the database. However, if the entered data fails validation checks, indicating errors or inconsistencies, the user receives an immediate warning notification. In response to the warning, the user is prompted to correct the input errors before finalizing the file upload process. This dynamic and user-friendly workflow ensures that only accurate and valid information is added to the application, enhancing the overall efficiency of resource management.

It is worth mentioning that the file download activity diagram is very similar to the File upload’s because it consists of mostly the same steps but in the reverse order, besides the parsing and validation which is not necessary for downloading.

2.1.2 Activity diagram for tracking progress

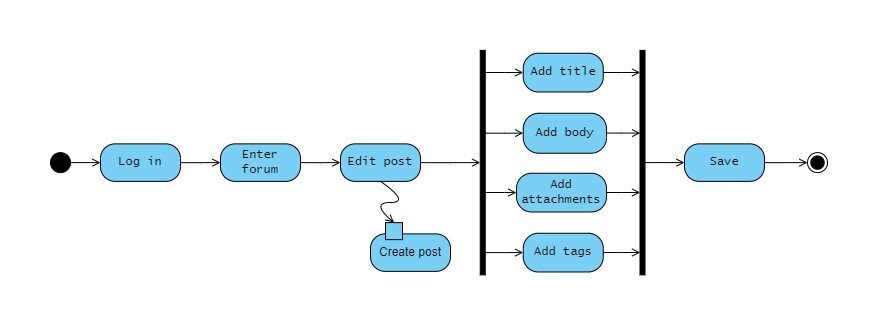


Figure 6 - AD02 – Forum usage activity diagram

The above diagram describes the forum usage activity, more specifically, editing a previously created forum post. As the activity itself suggests, the steps needed to contribute to the forum are purposely kept as simple, straightforward and few as possible in order to encourage communication between students.

2.1.2 Activity diagram for tracking progress

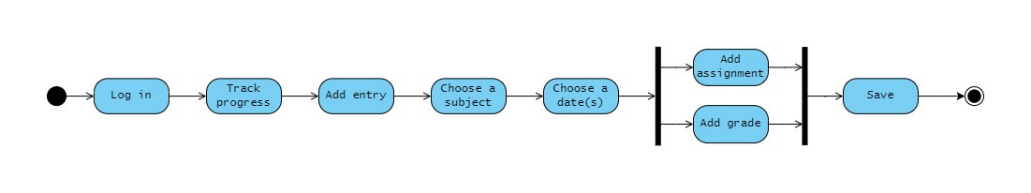


Figure 7 - AD03 - Progress tracking activity diagram

The above diagram describes the progress tracking activity. To initiate this process, users open the Progress page and choose the "Track Current Progress" action. Right after, the user is prompted to select one of the available subjects and a date or period to create an entry for. Finally, the user only has to input any additional information they want like a description or attachments.

These information allow users to, as the name implies, track the progress of each subject, later view them in the calendar and formulate a well rounded decision regarding their next steps.

2.2 Class Diagram

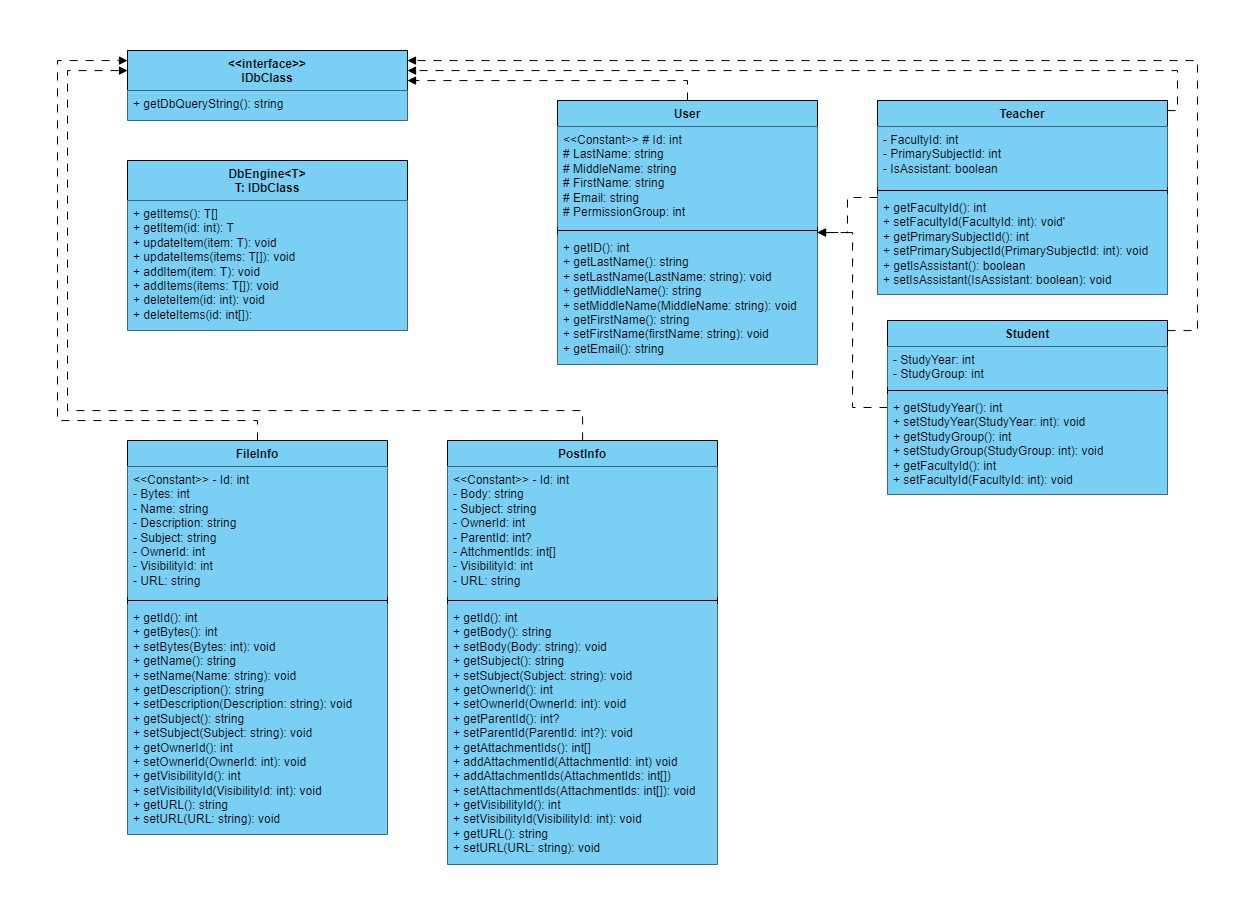


Figure 8 - CD01 - Class Diagram

In the application, there are a very large number of classes, but with the help of the class diagram [Figure 8] we can anticipate the pattern in which these are created. Firstly, the relationship between the User class and the two child classes Teacher and Student, perfectly exemplifies the way abstraction was used to facilitate development. CRUD methods are also available for most of the parameters in order to create a layer of encapsulation.   
 Secondly, the FileInfo and PostInfo classes are basically database models. These classes are auto generated using a database context which also gives us access to the database table name which we can later on use in the database engine class to write queries for these objects.

The databaseEngine class is a good example of how most of the engines work in the application; it is a generic class that takes advantage of the consistent structure of the model classes and their database self-referencing nature. Thus, we can create a DbEngine instance for any of the models and run the methods which will automatically build the necessary query string for the application. To be noted that this exact behaviour is also present for the App-to-API URL calls which are constructed in the same way, just with a different string structure.

2.3 State Diagrams

2.3.1 File entry state diagram



Figure 9 - SD01 - File entry state diagram

[TO BE CONTINUED]

2.3.2 Forum post state diagram



Figure 10 - SD02 - Forum post state diagram

[TO BE CONTINUED]

2.4 Interaction Diagrams

2.4.1 Add first year students sequence diagram



Figure 11 - SQ01 - Add first year students sequence diagram

[TO BE CONTINUED]

2.4.2 Generate class repartition sequence diagram



Figure 12 - SQ02 - Generate class repartition sequence diagram

[TO BE CONTINUED]

2.4.3 Track progress communication diagram



Figure 13 - CM01 - Track progress communication diagram

[TO BE CONTINUED]

2.5 Processes and collaboration diagrams in BPMN

2.5.1 Generate calendar view process diagram



Figure 14 - PD01 - Generate calendar view process diagram

[TO BE CONTINUED]

2.5.2 Forum discussion collaboration diagram



Figure 15 - CL01 - Forum discussion collaboration diagram

[TO BE CONTINUED]

3. Design of the system

[TO BE CONTINUED - INTRODUCTION]

3.1 Detailed Class diagram



Figure 16 - Application detailed class diagram



Figure 17 - API detailed class diagram

3.1.1 Class structures

[TO BE CONTINUED]

3.1.2 Application

[TO BE CONTINUED]

3.1.3 API

[TO BE CONTINUED]

3.2 Database design



Figure 18 - Database schema

[TO BE CONTINUED]

3.3 User interface design

3.3.1 Login



Figure 19 - UI - Login

[TO BE CONTINUED]

3.3.2 Main dashboard



Figure 20 - UI - Dashboard

[TO BE CONTINUED]

3.3.3 Progress



Figure 21 - UI - Progress

[TO BE CONTINUED]

3.3.4 Files



Figure 22- UI - Files

[TO BE CONTINUED]

3.3.5 Forum



Figure 23 - UI - Forum

[TO BE CONTINUED]

3.4 Component diagram



Figure 24 - CD01 - Component diagram

[TO BE CONTINUED]

3.5 Deployment diagram



Figure 25 - DD01 - Deployment diagram

[TO BE CONTINUED]