**ONLINE LIBRARY**

**APPLICATION REQUIRMENTS**

* **The application we will build contains a set of functions:**

1. Provides users with simple and advanced search options based on such search criteria as: author name, book title
2. Realizes a clear division of books into categories to facilitate search
3. Allows users to create and manage their personal accounts
4. Users can make online requests for various books
5. Informs users about books being added or deleted from the library
6. Offers the possibility of adding, deleting and updating existing books
7. Provides general descriptions of each book

* **User Features**

Since the online library is being built specifically to help the students, its users will definitely be the students of this university. The application will have a simple and flexible interface that greatly facilitates the search process, for this reason users do not need to have good technical knowledge except for basic knowledge to navigate the Internet. It will be aimed at creating a simple self-descriptive interface that will be comfortable and easily perceptible by users.

* **Restriction**

The application will have no hardware limitations, software limitations, security limitations or parallel processing. The only limitation will be the provision of a stable connection to the Internet for the use of its functionalities.

* **Functional Requirements**

In this section, all the functionalities of the software product will be described in detail based on the importance and performance level of each one. Each required function will be analyzed and presented through:

• Descriptions

• Criticism

• Technical issues

* **Simple and advanced search capabilities**

In this section the user can perform two types of searches. First, we have the simple search where the user can search for the title of the book, or its author, and all the books in our library will be displayed. Second I can perform an advanced search. In this type of search there will be several text boxes where the user will click or mark taking into account the requirements he has. So I can search for example for all the books of a certain category that have a price above a certain value, or that were written by a certain author

* **General descriptions of books**

The online library will contain all the books that are in the university’s physical library. After users complete their online order, they will be able to pick up the book at the physical library. The books will be in line with the existing branches of the university. So there will be books on economics, computer science, mathematics, statistics, etc. We have divided the latter into six major categories: Management, Economics, Marketing, IT, Accounting, Finance. Each of these categories has corresponding subcategories. Each book has attributes such as: title, author, a brief description of the book, and price.

* **Creating and managing user accounts**

Each user, student in this case will have his personal account so that he can use the functionalities of the application. The user creates the account the moment he tries to log in to the application. To register, he must fill out the form with data such as: first name, last name, place of birth, registration date, expiration date, email, phone number.

* **Online release of physical book requests**

After finding the desired book, the user can see the number of available copies of the book. If the number of available copies is different from zero, then he has the option to reserve the book by clicking the reserve button and physically appear near the library to retrieve the book. If the number of copies is zero, that is, the books are all reserved, then he clicks the borrow button and waits until the book becomes available.

* **Dividing books into different categories**

We have divided the books into six major categories based on the constituent branches of the university such as: Management, Economics, Marketing, IT, Accounting, Finance. Each of these categories has corresponding subcategories.

* **Adding and deleting different books**

The administrator has the right to delete or add different books. They can add new books as well as remove damaged books.

* **Performance Requirements**

In order to achieve its main purpose, the application supports simultaneous requests by processing them in parallel. The user does not have to wait more than 10 seconds to get the result of his search. Its flexible interface, with clear and distinct divisions, increases performance by facilitating navigation and reducing search time.

* **Design Constraints**

As we have mentioned, the application does not contain hardware or software limitations, nor in design. The design is built using client-side languages ​​such as HTML and server-side languages ​​such as PHP, which are easily configurable. Care must be taken in the unified use of the coding used and preliminary tests should be made as to how our application will look in different browsers: opera, chrome, firefox, etc.

* **Database logical requirements**

Our database will consist of 4 big entities: Category, User, Book, Authors. In addition to these, there will be another table which will keep track of which user received which book.

**DEVELOPMENT MODEL**

The development model that we have choose to work with is Aigle Model. That’s because this strategy works well for projects with flexible timelines that have changing requirements. The Agile methodology enables the incorporation of feedback at each step of development, guaranteeing that the finished product satisfies the requirements of its users. It involves frequent collaboration between our group members.