

what has learned in classroom to develop his software. In different semesters, this graduation course offers the most widely used programming languages, and C proved to be adequate for second semester.

Under weekly supervision of Development Laboratory professor, and to systematize its execution, case study process was divided into stages. The steps have been named as: Ideas, Tools, Modeling, Screens, BD and Final. An explanation of each step is presented next.

- Ideas

In the first step, learners would have to conduct a field survey to identify or generate possible needs that would be supplied by their software. This step demanded time and patience from the part of students and professors, because learners had to design an abstract idea of their software without knowing a programming language or database management system that would use. Professors fit the definition of software scope and encourage flowering of creativity and perception of each learner.

- Tools

In the second step, learners would have to think over some characteristics, such as whether software would be for web use, if there would be interaction between two different programming languages, what management system database to be used, among others. Professors gave students freedom to choose what they would like to work on.

- Modeling

In the third step, learner would do a survey requirement that the software should accomplish. From this, the learner could define entities and their attributes, generating entity relationship model for database.

- Screens

During the fourth step, student would have to present a prototype of software screens. In market, software that employs graphical user interface (GUI) is more attractive and interesting than others.

- Database

In the fifth step, student's challenge was to make connection between software and database, making possible to create, read, update, and delete on database.

- Final

In the last step, student should present the operating software and a scientific paper about development of his software. Though the paper will be not submitted to any event, its writing allows that the student organize his thinking.

During all steps, attitude of professors has been exercising an active and creative expects, offering alternatives for students believed they could complete the project and acquire meaningful learning.

5. Case study experiments

From each step of case study, byproducts arose such as skills, abilities, new ideas and solutions. Some of those results are shown next.

- Ideas

Every learner made a presentation about what would be his software and what need would be attended. Table 1 shows some of displayed items.

Table 1. Ideas software

Name	Problem	Goal
Projeto Dentista	Small dental offices do not have a computerized control of patient information.	To develop a software which allows storage and access to patient information.
ToDoList	People forget to perform simple activities during day and thereby impact implementation of major tasks and activities.	To deliver a software to manage daily tasks via web platform.
StockPocket	Multifunctional systems make difficult to control products.	To develop a simple and objective control system product.
Software de Repetição Espaçada	When people learn a new language often is difficult to keep words in mind.	To develop a web application which assists learners to review contents
Troca Fácil	To perform preventive maintenance of a car, technicians must search specifications referring to oil filter, air filters, fuel filters and lubricants. Those tasks take time.	To develop software to help technicians.

Table 1 presents five ideas which treat real problems. At this step, learners showed up insecure about completion of the projects. Professors defined scope of every software, guiding learners that the time for development was only one semester.

- Tools

Defined a focus for development, learners had to search and select tools which would be used in project performing. In general, choice fell on six groups of

tools: a) C programming language [14], interface development environment Code Blocks, creation environment Glade GTK graphical interface [15]; b) Interaction between the languages C # and C development environments Code Blocks and Visual Studio c) for Web development languages such as HTML, CSS and JavaScript; d) AngularJS [16] and JQuery framework; e) use of CGI (Common Gateway Interface) [17]; f) MySQL [18] was chosen as database manager system (DBMS) for all software.

On 1972, Dennis Ritchie developed C programming language at Bell Labs [14]. Table 2 presents how much C programming language is popular, among others.

Table 2. Ranking of programming language

July 2017	July 2016	Programming language	Ratings (%)
1	1	Java	13.774
2	2	C	7.321
3	3	C++	5.576
4	4	Python	3.543
5	5	C#	3.518
6	6	PHP	3.093
7	8	Visual Basic.NET	3.050
8	7	JavaScript	2.606
9	12	Delphi/Object Pascal	2.490
10	55	Go	2.363
11	9	Perl	2.334
12	14	Swift	2.253
13	11	Ruby	2.249
14	10	Assembly language	2.240
15	17	R	2.105
16	13	Visual Basic	2.096
17	16	MATLAB	2.009
18	15	Object-C	1.896
19	21	Scratch	1.813
20	18	PL/SQL	1.545

The above Table is adapted from [19].

Table 2 shows the top twenty ratings of programming language, and C is the second most popular in July 2017. According to [19], in March 2015, C programming language occupied the first position on ranking. The index shown in Table 2 is an indicator of the popularity of programming languages, which is updated once a month. Popular search engines such as Google, Bing, Yahoo!, Wikipedia, Amazon, YouTube and Baidu are used to calculate the ratings, which are based on the number of skilled engineers world-wide, courses and third party vendors [19].

Using C programming language, learners can write a code of program for computer. To write, compile and run this code, learner should use an

interface development environment (IDE). So, Code Blocks is a valid alternative of using, because it is open source, cross platform, free C, C++ and Fortran IDE [20].

As one of software’s requirements to be accomplished is the presence of graphical user interface, Glade GTK showed itself as an option. It is a tool to enable development of user interfaces for the GTK+ toolkit and the GNOME desktop environment. The Glade interface is saved as XML file and, by using GtkBuilder, this file can be used in C programming language, among others [15].

Some learners chose to integrate web pages and C program. In this case, the graphical user interface was made by using of JavaScript, HTML and CSS. To interconnect web pages and C program, learners used CGI, AngularJS and JQuery framework. CGI allows the computer to generate web pages instantly at the user's request [17].

MySQL delivers a robust SQL (Structured Query Language) database server, and it is a trademark of Oracle Corporation [18]. It has several interesting features, such as [18]: written in C and C++; works on many different platforms; provides the server as a separate program for use in a client/server networked environment; password security by encryption of all password traffic; support for large databases; and so on.

Every learner chose to use what tools they wanted. Professors just demanded C programming language should be included.

• Modeling

Every student-built entity relationship model of their software. That database should contain at least two related tables. From this step, each student began to study and to research items by on his own. Figure 1 shows logical model relating to TodoList database.

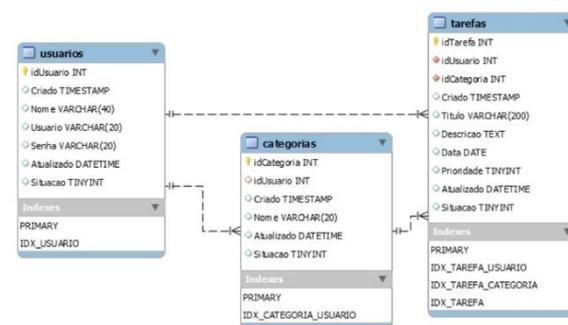


Figure 1. Logical model for TodoList

Figure 1 shows three tables named as usuarios, categorias and tarefas. Every table has attributes whose values are data to be stored. For example, from entity user, represented by table usuarios, seven data will be saved on database. Also, the relationships among the tables appears in Figure 1.

Building this model shown in Figure 1, student applied concepts of primary and foreign keys. Also, the type of every attribute appears. Each model was evaluated and corrected by Database Modeling professor. If there were any problems, student should work around.

- Screens
Development Laboratory professor showed some examples of screens, using Glade GTK and Allegro. From that, students built their prototype screens. An example is illustrated in Figure 2.



Figure 2. GUI of Projeto Dentista

Figure 2 illustrates initial screen of Projeto Dentista, which has a menu for patient’s registration and scheduling appointments. When modeling screen prototype, student had a more realistic notion of what would be his software and what could be developed. In this step, professors asked an interaction between screen and user, even partial.

- BD
After practical classes on SQL and basic database operations, students created their database and connected it to the software. Using C programming language, a student carried out the necessary code to access a MySQL database, running SQL commands. Figure 3 shows an excerpt of code related to those operations. As shown in Figure 3, listarCodPaciente function belongs to Projeto Dentista and executes a reading operation in the database. This function will return a list containing patient codes.

Also, Troca Fácil software executes queries operation to fill the fields motor power, year, air filter, oil filter, fuel filters and lubricants. According to selected car model, a list of items related to maintenance appears. Figure 4 exhibits a screen of Troca Facil software that illustrates the recovery of data on database.

```

106 char* listarCodPaciente(char cNome[])
107 {
108     char str[300];
109     sprintf(str, sizeof(str), "SELECT PCT_COD_PACIENTE FROM PCT_PACIENTE WHERE PCT_NOME='%s'", cNome);
110
111     if (mysql_query(con, str))
112     {
113         finish_with_error(con);
114     }
115
116     MYSQL_RES *result = mysql_store_result(con);
117     if (result == NULL)
118     {
119         finish_with_error(con);
120     }
121
122     int num_fields = mysql_num_fields(result);
123     int i;
124     MYSQL_ROW row;
125     while ((row = mysql_fetch_row(result)))
126     {
127         for (i = 0; i < num_fields; i++)
128         {
129             return row[i];
130             break;
131         }
132     }
133     mysql_free_result(result);
134 }
135
    
```

Figure 3. Sample of code

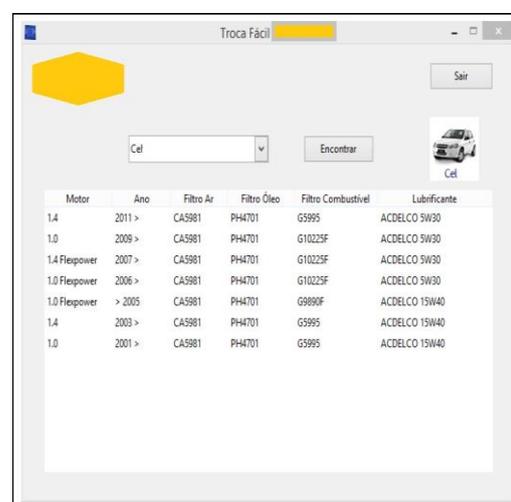


Figure 4. Recovery data on Troca Fácil

Figure 4 shows a screen where it is possible to visualize the image of car and all items of preventive maintenance. Looking at this screen, a mechanical technician can find the items for execution a correct maintenance on a car.

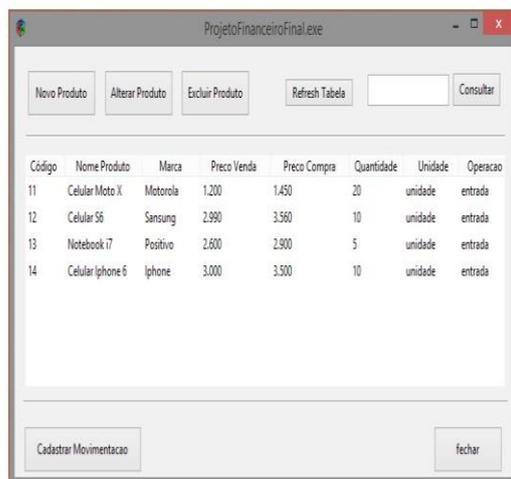


Figure 5. CRUD for StockPocket

Another example of queries operations on database is given in Figure 5. It shows results of CRUD on database. Figure 5 shows that from same screen of StockPocket software it is possible to accomplish CRUD (Create, Read, Update, Delete) operations. The buttons Novo Produto, Alterar Produto, Excluir Produto and Consultar make possible these operations.

Button Novo Produto allows to create a new product on database, storing all its data as code, name, trademark, purchase and sale prices, available quantity, unity and operation. The unity refers to which unity of measurement is used to count that product, e.g., inch, cm, km and so on.

To read data about a product, the user needs to provide its code and to click on button Consultar. From this, all data related to that product will be shown on the screen.

Whether the user wants to update data of any product, he need to click on button Alterar Produto. All data can be modified except the code of product because it is a unique identifier.

To delete a product of database, the user should click on button Excluir Produto. The execution of this operation means that the product will not be acquired by the company again.

On this step, all software is almost finished. The students adjust some details and professors provide the last concepts. Both professor and student feel that the journey was valid.

- Final

On the final step, students could think over their learning path, from beginning of semester to the delivery of software.

In this point, the writing of a scientific paper is recommended. Figure 6 illustrates a paper written by a student.

To direct writing of papers, Development Laboratory professor made available a template, based on [2], containing sections that should exist in an article, namely: Abstract, Introduction, Theoretical Framework, Literature Review, Proposed Solution, Description of the Developed Software and Conclusion.

In Abstract section, students should provide a concise description about the treated subject. The professor advised them that a good abstract declares the problem and its causes. In addition, the students were aware that an abstract should indicate the found solution or contribution for solving the problem, and its implications as well. Some abstracts, titles and number of pages of students' s papers are highlighted next

Title: Consultório Odontológico

Abstract: This article describes the development of Laboratory II material design. The encountered problem was the lack of a computerized system for dental offices that had no control of patient information. The solution was to develop an application that stores and manipulates patient's information. Pages: 6

Title: Software de Gerenciamento de Lista de Tarefas

Abstract: The idea of management to-do list software came from an issue found by the author in managing daily tasks, academics or not. Many times, we forgot of doing simple activities during the day and, for this reason, we ended up making difficult the implementation of much more important processes. Access to a to-do list will allow us gains of productivity in academic and/or professional areas, and, why not, others daily activities. Pages: 7

Title: StockPocket – Sistema de Controle de Estoque de produtos ou peças

Abstract: This article describes the creation of a simple and objective system of inventory control. There are many multifunctional systems hindering usability and jeopardizing the control of the products, the objective of this software is to walk the other way, with something basic and fully meet the functionality to which they relate. With a quick solution to this problem, the system seeks to achieve intuitive features like filters to streamline a seek registration, modification

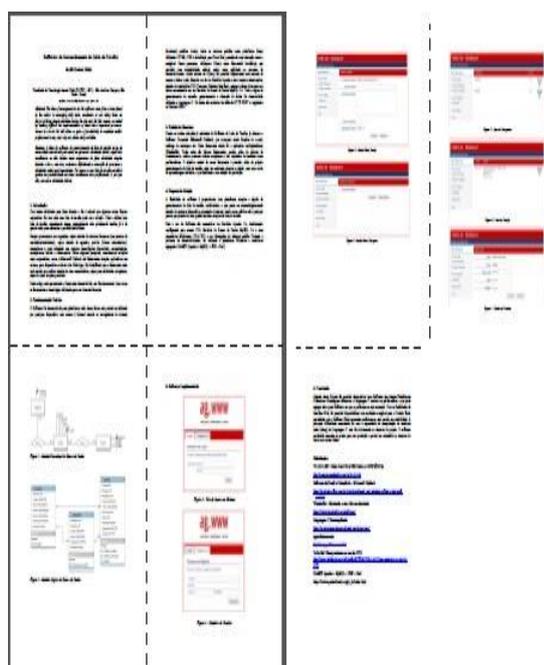


Figure 6. Sample of paper

and deletion of products and their respective drives.

Pages: 7

Title: Software de Repetição Espaçada

Abstract: The idea of spaced repetition software came from an issue found by the author in an English language learning journey. The observed problem was forgetting words. Studying more about it, the author found what researchers call Forgetting Curve which is basically the relationship between forgetting and time, the main idea is the person reviews the contents which he wants to memorize periodically, therefore making the time of forgetting increases exponentially.

Pages: 9

Title: Troca Fácil XX

Abstract: This article describes the solution to an everyday problem in the situation where the machine shop technician to carry out consultation with oil filters, air filters, fuel filters and lubricants for motor vehicles the automaker XX, do not waste time doing manual queries in paper catalogs, thus saving time and avoiding problems of carrying out the exchange for any incorrect piece. For the Troca Fácil XX System will assist in this step.

Pages:11

The presented abstracts reveal a little more about the writing style of students, their ideas, observed problems and their designed softwares. Though improvements and revisions are necessary, the students could write their first papers.

In Introduction section, the professor said to her students to try answer the following questions:

- a) Why this issue is interesting?
- b) What is the main involved problem on?
- c) Is there some previous solution for the problem?
- d) What are proposal of solution and objectives?
- e) What is the focus of the work?
- f) What contents are going to be presented by next sections of the article?

In Theoretical Framework section, the students should describe the theory about the subject and which matters they needed to study for implementation of software. In addition, Literature Review section should present related works, i.e., similar softwares.

Proposed Solution section should explain all details of implementation and design of software. In Description of the Developed Software section, the students should present a brief tutorial of their software.

Among all sections of every article, conclusion was the most important because it described what student learned from the project and encountered

difficulties during development. From conclusion, professors could extract materials and ideas to evaluate and improve proposal of interdisciplinary project. Some excerpts of students' s conclusions are highlighted next.

"At the beginning of development, to learn in C programming language was difficult due to its syntax and structured programming mode (...). The greatest difficulty was to connect and manipulate both C programming language and Glade Gtk graphical interface, due to the lack of materials available for study in Portuguese. This was solved with the help of the teacher who provided contents and tips for creating and interacting Glade components with the C language programming. In the future, I intend to continue the project making it more complete in relation to the management of dental offices."

Student – author of Projeto Dentista.

"Working in this project it was possible to develop a software that integrates different platforms and paradigms. The C programming language has shown good performance (...). The use of web interface provided a friendly environment for end user. The main difficulty was with the ability to manipulate text variables (string) on C programming language. This issue was solved during the project."

Student – author of TodoList.

"During the development of this project, new learning, new techniques for problem solving, (...) and so on were acquired. First, the encountered difficulties were to connect graphical interface Glade GTK, database MySQL and to understand how pointers work. Understanding how pointer work helped a lot in the development of the program, as well as learning about the communication of widgets. The system can be concluded due to guidance of teachers, self-studies, and knowledge acquired from several subjects of the course (...)"

Student – author of StockPocket.

"The main difficulty was the manipulation of texts on C programming language, which was solved with research and with the absorbed content during the lessons (...)."

Student – author of Software de Repetição Espaçada.

"It was possible to learn a new programming language, new problem-solving techniques, a new logical thinking format. Besides, we understood

that dedication makes learning flows. During development, there were many difficulties, such as creation of interface screens, communication with the database and interconnection among the languages. All problems were solved with class follow-up, teachers' orientation, studies and research(...). "

Student – author of Troca-Fácil.

The act of writing a conclusion led students to thinking over all their work, including successes and challenges. The students could review their learning path and observe their progress.

6. Conclusion

This paper presented an interdisciplinary proposal applied into a case study. Undergraduate students developed inherent skills and abilities to researchers. From this case study, positive outcomes were collected.

Students took an active attitude towards teaching-learning process, their self-esteem increased, they realized they can design and develop something of merit. Besides, they realized connection between course subjects and importance of each one.

Students mentioned apprenticeship of new technologies and languages as positive points. Also, they cited scientific writing as a discovery for them. Others said to learn something that will have immediate application in a software designed and developed by themselves is much more exciting.

Another interesting point was students said they felt to improve English learning is needed because many manuals and materials of new technologies are written in that language. Also, to write abstract of paper, students had to consult English professor to remedy doubts. Indirectly, subjects related to communication and expression have benefited from realization of interdisciplinary project.

Students stressed that learned to seek solutions for problems, researching unknown subjects and to develop a new format of thinking and logic. In addition, students reported teaching-learning process flows more naturally, when dedication and work exist.

Using an interdisciplinary approach, teaching paradigms can be reviewed, contributing to emerging of new researchers among many students. That initiative will provide the flourishing of more innovation and development in country.

Professors concluded that to join efforts produces better outcomes, and because that, they intend to extend the experience even more.

7. References

[1] P. Piazzzi, *Aprendendo inteligência: manual de instruções do cérebro para alunos em geral*, 2nd ed. São Paulo: Aleph, 2008.

[2] Raul Sidnei Wazlawick, *Metodologia de Pesquisa para Ciência da Computação*, 2nd ed. Rio de Janeiro: Elsevier, 2014.

[3] M. Frické, "The knowledge pyramid: a critique of the DIKW hierarchy," *J. Inf. Sci.*, vol. 35, no. 2, pp. 131–142, 2009.

[4] A. Da Silva Jacinto, T. M. Almeri, E. Lamas, L.A. V. Dias, and A. M. Da Cunha, "Computer science and interdisciplinarity: A case study on an undergraduate program," *Proc. 2013 10th Int. Conf. Technol. New Gener. ITNG 2013*, pp. 289–293, 2013.

[5] I. R. Peleias, J. de F. Mendonça, V. G. Slomski, and I. C. A. Fazenda, "Interdisciplinaridade no ensino superior: análise da percepção de professores de controladoria em cursos de ciências contábeis na cidade de São Paulo," *Avaliação, ...*, vol. 16, no. 3, pp. 499–532, 2011.

[6] A. M. N. Machado and L. Bianchetti, "(Des)feticização do produtivismo acadêmico: desafios para o trabalhador-pesquisador," *Rev. Adm. Empres.*, vol. 51, no. 3, pp. 244–254, 2011.

[7] C. M. Haas, "Interdisciplinaridade: Uma Nova Atitude Docente," *Olhar Profr.*, vol. 10, no. 1, pp. 179–193, 2009.

[8] F. Hacklin and M. W. Wallin, "Convergence and interdisciplinarity in innovation management: a review, critique, and future directions," *Serv. Ind. J.*, vol. 33, no. 7–8, pp. 774–788, May 2013.

[9] J. da S. Thiesen, "A interdisciplinaridade como um movimento articulador no processo ensino- aprendizagem," *Rev. Bras. Educ.*, vol. 13, no. 39, pp. 545–554, Dec. 2008.

[10] A. da S. Jacinto, T. M. Almeri, and J. M. P. de Oliveira, "Aprendizagem baseada em problemas mal-estruturados: Programação e Cidadania," in *Anais do XXII SBIE - XVII WIE*, 2011, pp. 1197–1206.

[11] J. D. A. Lopes, "Práticas Interdisciplinares No Ensino Superior: Dilemas E Estudos Nas Áreas De Letras E Artes Interdisciplinary Practices in Higher Education: Dilemmas," no. 1, pp. 77–86.

[12] E. A. Araújo and J. de Lopes, "As práticas Interdisciplinaridades: Análise dos obstáculos didáticos e epistemológicos das áreas de Ciências Humanas e Ciências Sociais Aplicadas," *Rev. Educ. PUC-Campinas*, vol. 15, pp. 89–98, 2003.

[13] M. Marzano, D. N. Carss, and S. Bell, "Working to Make Interdisciplinarity Work: Investing in Communication and Interpersonal Relationships," *J. Agric. Econ.*, vol. 57, no. 2, pp. 185–197, Jul. 2006.

[14] A. Backes, *Linguagem C: Completa e Descomplicada*. Rio de Janeiro: Elsevier, 2013.

[15] Glade -A User Interface Designer, *The Glade Project*, 2014. [Online]. Available. <https://glade.gnome.org>, (Access date: 10 Apr, 2015).

[16] Build Apps with AngularJS, Chrome, [Online]. Available: https://developer.chrome.com/apps/angular_framework#_blank, (Access date: 12 Apr, 2015).

[17] S. Gundavaram, CGI Programming on the World Wide Web, 1st ed. O'Reilly Open Books Project., 1996.

[18] MySQL, Oracle, 2017. [Online]. Available: <https://www.mysql.com/>, (Access date: 05 May, 2015).

[19] TIOBE - The Software Quality Company, TIOBE software BV, 2017. [Online]. Available: <https://www.tiobe.com/tiobe-index/>, (Access date: 12 Jul, 2017).

[20] Code: Blocks - The IDE with all the features you need, having a consistent look, feel and operation across platforms, Code: Blocks, 2017. [Online]. Available: <http://www.codeblocks.org/>, (Access date: 12 Jul, 2017).