An Example of Designing Modular Vocational Syllabi

Panagiotis S. Makrygiannis, A. Paradisi, T. Tsapelas, Evangelos C. Papakitsos, and Dimitrios Piromalis

***Abstract*—In this paper, an application in developing a course for secondary vocational education settings in a modular manner is discussed and a number of conclusions are drawn from the process. The application was partly intentional and partly the result of circumstances, as it was realized, being a part of the redesign process of the specialties offered within the sector of Informatics of Vocational Lyceums (upper-secondary education), the schools that offer initial vocational education in Greece. Using modular design is applicable to secondary vocational education courses. It allows for adaptability, reusability and a variety of approaches within a competence-based syllabus. It is also suggested that more courses could benefit from a modular structure, definitely including other advanced laboratorial courses, but possibly even introductory ones.**

***Index Terms*—Curricula, Modular Syllabus, Vocational Education.**

I. Introduction

Modular curricula and syllabi have been proposed as a solution for the demand/offer deficit of higher education and training and a way to alleviate the pressures put on academic staff, through increased teaching loads, while they also need to develop and strengthen their research profile for more than a decade now [1]. They are also tightly connected with the competence-based curricula paradigm [2]-[4] and credit introduction [5] in both tertiary education and training.

At the same time, some applications of modularization have been taking place in secondary education settings, with most notable example the modularization of subject criteria for the General Certificate of Secondary Education examinations in England [6]. Since September 2009, almost all specifications are modular in structure, with Mathematics, English and Science paving the way. While this instance is mainly focused in modular specifications, exams and certification, others are more focused on organizing content with reference to high and even middle-school [7].

Modular design has been related with distance [8], [9] and blended learning [10] lessons, with Open curricula [11] and in connection with reusability [12]. Still, modular design is not without its pitfalls and requires attention in application [1], [4], especially when applied indiscriminately [6]: it seems that, in language lessons, results are not guaranteed, although application in such lessons is quite frequent and not at all new, while in math and science the results seem more assured. The attempt to ensure effective learning, from modular courses, go back at least 30 years ago [2], [13], and are yet to conclude [12], [14].

In this paper, an application in developing a course for secondary vocational education settings in a modular manner is discussed and a number of conclusions are drawn from the process. The application was partly intentional and partly the result of circumstances, as it was realized, being a part of the redesign process of the specialties offered within the sector of Informatics of Vocational Lyceums (upper-secondary education), the schools that offer initial vocational education in Greece.

II. A Modular Syllabus

A. The Context

According to the Study Guide of Vocational Lyceum [15], the initial vocational education in Greece is delivered in three years, with:

- the first one being introductory;
- the second one, introducing sector-specific courses and material common to all specialties, offered within the relevant sector;
- the final year, introducing further specialization, depending on the specialty.

The sectors include various technical and vocational groupings, as for example mechanical engineering, nautical, informatics, agricultural, economy and administration, paramedical etc. In the case of Informatics sector, there are two specialties offered since 2017: ‘Information Applications Technician’ and ‘Personal Computers and Networks Technician’.

Studying in Vocational Lyceum allows for entering tertiary education in specialty relevant fields of study, via nationwide exams in four courses. Two of them, Language and Mathematics, are common for all graduates and two of them are technical and/or vocational, specified by the specialty. In the case of Informatics sector, these two technical courses are common for the two specialties, titled Programming and Networks.

At the same time, the Vocational Lyceum offers vocational qualification for each specialty and measures were taken to be relevant to the labour-market’s needs. In a preliminary study, initially conducted by one of the authors, then a Consultant at the (Greek) national Institute of

Published on February 17, 2020.

P. S. Makrygiannis is with the Department of Industrial Design & Production Engineering, University of West Attica, Egaleo, Greece (e-mail: mgiannis@uniwa.gr).

A. Paradeisi is with the 2nd Vocational Lyceum of Ilion, Ilion, Greece (e-mail: artparad@gmail.com).

T. Tsapelas is with Filothei Lyceum, Filothei, Greece (e-mail: stsap@gmail.com).

E. C. Papakitsos is with the Department of Industrial Design & Production Engineering, University of West Attica, Egaleo, Greece (e-mail: papakitsev@uniwa.gr).

D. Piromalis is with the Department of Industrial Design & Production Engineering, University of West Attica, Egaleo, Greece (e-mail: piromali@uniwa.gr).

DOI: http://dx.doi.org/10.24018/ejers.2020.0.CIE.1794

Educational Policy, there were evidence of constant demand for both experts and experienced users in the fields of:

- relational databases,
- web applications and
- mobile applications,

among other things. For reasons related to the division of the curriculum to syllabi some material on web applications found itself in the material of our lesson of interest on mobile applications.

The course in question, titled Special Issues in Computer Programming [16], is now taught in the third grade of Vocational Lyceums in Greece. Originally, it was conceived as two separate courses, titled:

- Special Issues in Computer Programming for the Web and
- Special Issues in Computer Programming for Mobiles,

The courses corresponded to two separate specialties within the Informatics Sector. The two courses eventually converged into a single one, when, in designing their syllabi, it was realized that their common stem was proven to cover most of their teaching hours. The two specialties also gradually lost their distinctiveness as time passed, becoming one as of 2017.

In the weekly teaching schedule of this specialty nowadays, four (4) teaching hours are allocated to Special Issues, all of them in the computer laboratory. Originally, the concept was of one hour of lecture and three hours of laboratory time, but most of the material, supposed to be included in the lectures, could also be handled in a more connectivistic, socially embedded, group learning approach. Allocating all four teaching hours to a laboratory keeps options open. After all, if computer engineering students, throughout their education, should encounter different approaches to design so that they become familiar with the strengths and weaknesses of these approaches [17], the same should apply to some extent to technical professionals.

The choice of keeping the whole course in a laboratory has a few drawbacks:

- Not all laboratories are recent procurements and thus some are not capable of accommodating modern computing environments. This is one reason for the existence of Module 2B (see subsection *D Part 2: Apps Development for Android*).
- There is also the fact of the variety of backgrounds of the secondary education teachers of Informatics that might play a role in the applicability of the original design, as well as the initial level of laboratory readiness among the students.

Therefore, the two modules of the second part have both a short and an extensive version to be used alternatively, depending on the teacher/class/laboratory combined conditions.

B. *The Content*

As a result of the previous process, the syllabus presented is about an introduction to:

- object-oriented programming, on one hand, presented via the Java programming language, and
- the development of integrated applications, on the other, using the Android platform applications as an example.

Java is a programming language that can support the development of any application, on any platform, from personal computers and Android phones to digital television, either stand-alone or web-based. The short development time of an application, especially from inexperienced developers, is among the merits of the language. Taking advantage of this Java feature and the available programming environment, the students at the end of the course should be able to design and implement integrated small-scale applications, with an object-oriented philosophy.

The vocational aspect of the specialty is what the syllabus of the course best serves, developing skills and competences in both areas. Since the technological aspect of both web and mobile applications is somewhat on the forefront of technological evolution, right from the first draft of the syllabus it was obvious that even the most cutting edge choices of technology might turn out to be obsolete, in a relatively short amount of time. Thus, we were confronted with the need for a modular structure that would allow for both adaptation, through partial replacement, and reusability for the rest of the material.

The syllabus is structured into two parts, with each part consisting of two modules of several chapters. Specifically:

- the first part on object-oriented programming (see next subsection) consists of eight (8) relatively hefty chapters, while
- the second part on developing Android applications (apps) consists of 20 shorter chapters (see subsection *D Part 2: Apps Development for Android*).

The difference of sizes between chapters of the first part and chapters of the second will be justified, once the difference in approach between the two parts is made obvious.

C. *Part 1: Object-Oriented Programming*

The tools used in Part 1 of the course are essentially two:

- The GreenFoot programming environment and
- Eclipse.

Initially, advantage is taken of the educational benefits of GreenFoot (chapters 1-3), for a first introduction to object-oriented language features and event-oriented programming. The presentation of algorithmic selection and iteration structures is brief, as it is a teaching prerequisite in a previous introductory course, either in pseudo-language or in another programming language (Python). This is module 1A and represents a basic introduction to Java.

It is suggested that the main programming environment of the course should be Eclipse, which has all the application development tools integrated, e.g., Javadoc, Ant, JUnit. However, if it is considered too demanding for a particular computer lab, we can use a very light source code editor such as Notepad++, although compiling and running applications in this case should be done from the command line, which is not particularly attractive to students. The teacher can use any Java programming environment that allows him/her to achieve the curriculum objectives. It is also suggested to use the ACM Java Task Force (JTF) library, which will initially help us to avoid the difficulties encountered, when programming input/output functions. Using this library will help us conceal many insignificant details of the language, which in the first lessons may be quite difficult for students. This is module 1B and represents an in-depth familiarization with Java, as a language for developing applications and object oriented, event driven programming.

It should be noted that the general purpose is to get students familiar with the philosophy of developing an integrated Java application, rather than learning the details of a package or programming environment that may be outdated tomorrow. So, as they develop their application, it is a good practice to search the web for the best solutions to problems they encounter, through community sites, such as stackoverflow.com, and to be able to use the documentation of Java libraries to quickly and easily find whatever they are looking for.

It is suggested that the projects be prepared by groups of 3-4 students. The role of the teacher is crucial here. It should guide students in the division of their tasks, e.g., one student can undertake the design of the graphical interface, another one the communication with the database, etc.

In each chapter, there should be several examples of code that students need to perform in the lab and investigate their results. A good practice would be to give students the necessary code files to build the app and experiment with it. The aim is to discover, with the guidance of the teacher, the basic characteristics of the language. The programs presented in each chapter could be modified or extended to include more general cases.

D. Part 2: Apps Development for Android

The second part aims to familiarize students with the development of integrated applications for the Android platform. This platform is mainly concerned with mobile devices and tablets, where its role in the market is significant. It has also significant features as it not only allows but also facilitates communication between applications, utilizing parts of one another and sharing resources. If the conditions allow the knowledge gained in Part 1 to be used here, this would be ideal.

In any case, the teacher is advised to make use of the discovery learning notion, both to discover the details of the application that have been deliberately omitted and to extend it, starting with possible suggestions from the students. The development of a parallel application of their choice could allow for a significant deepening of students' knowledge through the implementation of exemplary applications of the two parts.

Modules 2A and 2B of Part 2 form a modular assembly. Two alternative scenarios are provided for their teaching:

- In the first one, 2A covers 12 hours of lectures and 34 hours of laboratory training, while 2B covers 1 hour of lecture and 13 hours of laboratory training. Module 2A is about developing applications using deployment packages such as Eclipse or Android Studio and utilizing Java and XML. Module 2B is about developing applications in the AppInventor novice programming environment, which uses tile programming and is taught in a demo format by the teacher.
- In the second scenario, which applies only if the laboratory conditions do not allow the installation of the Android platform (Android development package such as SDK and ADT combination with Eclipse or Android Studio), module 2A (11 hours of lectures) is presented as a demo by the teacher and the rest available teaching hours are devoted to module 2B (2 hours of lectures and 47 hours of laboratory learning).

Module 2A concerns, in its broad form, the development of an application within a school laboratory. It thus has the rationale of supporting material, to be used in a school laboratory, in the presence of teachers, in the context of a supported laboratory practice. It does not have the rationale of standalone material for individual study or a book that exhausts the anyway applied subject. A repetition of some parts has the sense of a Bruner spiral, where the second appearance is combined with greater depth (in the first one, the demonstration; in the second one, the implementation).

Module 2B has the explicit purpose of developing a modular application in the AppInventor tile programming environment. It combines the development ease of this environment with the rich capabilities of the Android platform, to help students understand the functionality of a complex application. This module is an integral part of Part 2, concerning Android. In its extended form, it is combined with sections of the theory of module 2A, in order to provide the framework for introducing the relevant concepts.

Ideally, this module will take the minimum of the two alternatives and will either precede 2A, to create a frame of reference, or follow it, in order to assimilate the key concepts into practice, depending on the capabilities of the schools' labs. The latter is not necessarily negative, as almost the whole process of developing an Android application can be presented in that context. In this case, it will be necessary to incorporate parts of theory from module 2A into the teaching of this module. It should also point out the parts of developing an application that the environment automates to a large degree and those to be examined externally by it.

It is clear that module 2A has more to offer to the training of professionals in the labor market. The ability to easily obtain an integrated view of the application development process, as presented in module 2B, has also significant educational benefits. The choice is left to the teachers and depending on the labs available in their schools.

E. Commentary

Initially, the choice is to exploit the capabilities of Android SDK to combine with the Eclipse environment, which appeared in the Java programming section, using Android Development Tools (ADT). The resources needed to develop the Android application on the computer will be created and fully functional applications will be created as well.

However, it should be borne in mind that Google's support for ADTs in Eclipse has been discontinued and the focus is now on the new Android Studio (AS) development platform. The course should, therefore, be adapted to the

DOI: http://dx.doi.org/10.24018/ejers.2020.0.CIE.1794

new reality, when the new environment becomes sufficiently stable and homogeneous to replace the existing one. Fortunately, AS also uses the Android SDK, making the transition relatively easy.

Along the way, it will be clear that Android is a completely different case than Java, because it is not, nor does it include, a programming language. This means that developing applications for this will also require some computer programming in a programming language. Fortunately, much of the programming can be done in Java. Another part works well with a declarative page markup language, namely XML, whose essentials are presented wherever needed. Alternatively, implementation can be done entirely in the tile programming language of the AppInventor novice programming environment, with the restrictions that such a language imposes.

III. Conclusions and Directions for Research

Rapid technological advancements and subsequent changes in educational and training needs and habits have given a renewed rise to the demand for constant renewal and thus modularity in curricula. Distance and blended learning and both the demand for and, lately, the abundance of open resources also plays a part in it. The need is well documented [14], [18] and fuels the discussion for higher education reform.

Using modular design is applicable to secondary vocational education courses too. It allows for adaptability, reusability and a variety of approaches within a competence-based syllabus. Competence here is understood to refer to the performance standards, associated with a profession or membership to a licensing organization [19]. The modules do not need to be of the same size, although our use of the term tends towards more substantial sizes than sometimes used in on-line education.

While modular specifications, exams and certification were not a major consideration, some catering for modular assessment is possible. The provision of a mid-term exam, combined with the possibility of partial deliverables (easily attainable while working with a project-based learning approach), seem to serve well such a purpose. Furthermore, projects ensure that graduates can successfully apply the knowledge that they have gained, and if all graduates of computer science programmes should have been involved in at least one substantial project [20] then perhaps the same should apply for technicians.

Some teachers have reported back to the coordinators of the course's syllabus that the course is somewhat demanding but also gratifying. That seems to be the case especially when very experienced or still in contact with the private sector teachers decide to take up the course. A possible educational research project could be designed to either verify those reports or record different stances, among the secondary education teachers involved.

It seems to the authors that the points made in this section suggest that more courses could benefit from a modular structure. That could definitely include other laboratories but possibly even introductory ones. Some thought should be given in that direction, before the next redesign of the Informatics specialties is conducted.

Acknowledgment

The authors would like to thank the Postgraduate (MSc) Programme of Studies "Industrial Automation", of the Dept. of Industrial Design & Production Engineering of the University of West Attica, for providing the required resources and funding in order to undertake this research project.

References

[1] R. Donnelly, and M. Fitzmaurice, "Designing Modules for Learning," in *Emerging issues in the practice of University Learning and Teaching*, G. O'Neill, S. Moore and B. McMullin, Eds. Dublin: All Ireland Society for Higher Education (AISHE), 2005.

[2] I. R. Cornford, "Ensuring effective learning from modular courses: a cognitive psychology-skill learning perspective," *Journal of Vocational Education & Training*, vol. 49, no. 2, pp. 237-251, 1997.

[3] C. Keane, and H. Houlton, "Improving Workforce Readiness Through Competency-based Education Modules," presented at the 19th Annual Sloan Consortium – International Conference on Online Learning, Buena Vista, Florida, USA, November 20-22, 2013.

[4] R. G. Sakhieva, E. R. Khairullina, L. G. Khisamiyeva, N. S. Valeyeva, A. R. Masalimova, and V. G. Zakirova, "Designing a Structure of the Modular Competence-Based Curriculum and Technologies for Its Implementation into Higher Vocational Institutions," *Asian Social Science*, vol. 11, no. 2, pp. 246-251, 2015.

[5] M. Betts and R. Smith, *Developing the Credit-Based Modular Curriculum in Higher Education*, London: Routledge, 1998 (republished as an ebook, August 2005).

[6] C. L. Vidal Rodeiro, and R. Nádas, "Effects of modularization," Cambridge Assessment, Assessment Research and Development, Research Division, 2009.

[7] D. Thalkar, Z. Blumberg Corwin, E. Swensen, and S. Bouchard (2015). Future Bound, Modular Curriculum for Middle School Teachers [Online]. Available: http://pullias.usc.edu/wp-content/uploads/2015/10/FutureBound-Curricular-Modules.pdf

[8] J. Jorstad, and R. Hoar, "Transforming an Online Learning Module Into a MOOC," presented at the 19th Annual Sloan Consortium – International Conference on Online Learning, Buena Vista, Florida, USA, November 20-22, 2013.

[9] J. Rhode, "Designing Self-Paced Training Modules as the Foundation of Online Faculty Certification," presented at the 19th Annual Sloan Consortium – International Conference on Online Learning, Buena Vista, Florida, USA, November 20-22, 2013.

[10] E. Z. Rukobo, and A. Paul, "Design Principles for Competency Based Components in a Blended Learning Model," presented at the 19th Annual Sloan Consortium – International Conference on Online Learning, Buena Vista, Florida, USA, November 20-22, 2013.

[11] A. W. Bates (2015), *Teaching in a Digital Age*. Geneva: University of Geneva [Online]. Available: http://tecfa.unige.ch/guides/e-books/Bates-Teaching-in-a-Digital-Age-compressed.pdf

[12] J. M. Lightfoot, "Modular Curriculum Design Using Personal Learning Plans and Reusable Learning Components," *Communications of the IIMA*, vol. 6, no. 4, pp. 65-80, 2006.

[13] P. J. Van Eijl, "Modular programming of curricula," *Higher Education*, vol. 15, no. 5, pp. 449-457, 1986.

[14] K. Carey, *The End of College*, New York: Riverhead Books, 2015.

[15] A. Oikonomou, N. Fotiou, K. Parastatidis, and K. Kaltsas, *Study Guide of Vocational Lyceum*, Katerini, Greece: free – e book, 2018 (in Greek).

[16] E. Vrachnos, I. Kouretas, P. Makrygiannis, and A. Paradeisi, *Special Issues in Computer Programming*, Patras & Athens, Greece: "Diophantus" Computer Technology Institute and Press, 2016 (in Greek).

[17] *A joined Report on Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering*, Association for Computing Machinery (ACM) and IEEE Computer Society (IEEE-CS), 2016.

[18] *Institute-wide Task Force on the Future of MIT Education*, FMITE Final Report, 2014.

[19] *A joined Report on Curriculum Guidelines for Baccalaureate Degree Programs in Information Technology*, Association for Computing Machinery (ACM) and IEEE Computer Society (IEEE-CS), 2017.

[20] *Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*, The Joint Task Force on Computing Curricula Association for Computing Machinery (ACM) and IEEE Computer Society. (2013).

DOI: http://dx.doi.org/10.24018/ejers.2020.0.CIE.1794

Panagiotis S. Makrygiannis was born in Athens in 1968. He graduated from the Dept. of Informatics, National & Kapodistrian University of Athens, Greece, in 1995. He holds a European Master in Multimedia & Audiovisual Business Administration issued by Haute Ecole St. Louis, Brussels, in 2000, and a Master of Education (M.Ed.) in Educational Technology and Development of Human Resources issued by University of Athens in 2010. He also earned a postgraduate diploma in Vocational Guidance and Counseling from Panteion University of Social & Political Sciences, Athens, Greece, in 2008.

He has 20 years of teaching experience in secondary vocational education and has worked as sessional staff in three tertiary education institutions in Athens, Greece. He has served in positions of scientific, pedagogic and administrational responsibility. He has conducted research for various scientific and professional organizations and coordinated the Inter-Scientific Institute for European Strategy, Analysis and Planning for a number of years (2002-2013), Author, co-author and editor of a number of books on the application of IT in various sectors and also teaching books for students of informatics, has also been a member of Syllabi development committees of the Institute of Educational Policy of Greece for the corresponding lessons. He is currently conducting research on extended blended learning and teaching technologically heavy subjects as part of his PhD studies with the University of West Attica.

Mr. Makrygiannis is a member of various professional and scientific organizations and served as a member of the Board in several of them. He is currently serving as President of the Board of the Scientific Association for Technological Education and Training.

A. Paradisi was born in 1964. She graduated from the Dept. of Computer Systems Engineering, Technological Educational Institution of Pireas in 1989. She holds a Master of Science and Engineering in Applied Computer Science issued by VRIJE University of Brussels in 2011.

She has served in the Department of Informatics of “Evangelismos” Hospital (1992-2002) and after that (2002-2014) in the 2nd Vocational Lyceum of Ilion in the capacity of a teacher of Informatics. At the time of the design of the Syllabus presented here (2014-2017) she was in secondment at the national Institute of Educational Policy. She currently serves at the Special Division of the Integrated Information System of the Ministry of Development in Athens, Greece.

Mrs. Paradisi has contributed in the ICERI2011Proceedings with the article A. Paradisi, M.P. Uwase, J. Tiberghien, K. Steenhaut, J.M. Dricot (2011) Didactic Simulators for understanding Routing Protocols in Wireless Sensor Networks. She is also co-author and editor in two teaching books for students of informatics and also a member of the Syllabi development committees of the Institute of Educational Policy of Greece for the corresponding lessons. One of those lessons is the subject of the current paper. Mrs. Paradisi is member in various professional societies.

T. Tsapelas (Dr.) is a secondary education teacher of Informatics. He holds a PhD in Reliability.

He has served in various positions of responsibility in the public sector and has been involved in a number of operational programmes including that of “Digital Convergence 2007-2013”. At the time of developing the new curriculums for students of Informatics in Secondary Vocational Education he was serving as a Consultant at the (Greek) national Institute of Educational Policy on the subject, in effect having the responsibility for their structure and development. He is currently teaching Informatics in Filothei Lyceum.

Dr. Tsapelas is a member and has served on the board of Greek Computer Society, the PanHellenic Union of Information Teachers and other scientific bodies.

Evangelos C. Papakitsos was born in Athens, in 1963. He graduated from the Dept. of Physics, National & Kapodistrian University of Athens, Greece, in 1987. He earned an MSc in Information Systems from The University of Liverpool, UK, in 1992 and a PhD in Linguistic Computing from the National & Kapodistrian University of Athens, Greece, in 2000. He also earned two postgraduate diplomas in Vocational Guidance and Counseling from the Athens University of Economy & Business, Greece, in 2000 and from Panteion University of Social & Political Sciences, Athens, Greece, in 2008.

He has served for two years as a Greek Army Officer (now in Reserve) and worked for 27 years as a secondary education teacher in Greece (public appointment). He has also worked as academic staff in four tertiary education institutions of Athens, Greece, now being a member of the academic staff of the University of West Attica. He is the author or co-author of more than 120 articles and 29 books, studies and manuals.

Dr. Papakitsos is a member of various professional organizations and he has participated in several committees of the Hellenic Ministry of Education.

Dimitrios Piromalis (Dr.) is Assistant Professor in the field of “Design and Optimization of Hardware and Hardware Wirelessly Built-in Embedded Systems” at the University of West Attica. He organizes and teaches undergraduate and postgraduate courses and supervises a group of doctoral candidates. His scientific work includes more than 100 articles, while his writings include publications on tertiary education and industry with topics on Data Collection Systems, Electrical Circuits, Micro Controllers, and more. He has an over twenty-five years of parallel experience in the private sector, in designing new product-systems for the Greek electronics industry, in areas such as energy management, precision agriculture, smart cities, automotive systems, home automation, health and athletic systems, as well as industrial systems. His research background includes a number of research and development projects on topics of innovative systems designing for the industrial, domestic, shipping, medical, rural and automotive systems.

DOI: http://dx.doi.org/10.24018/ejers.2020.0.CIE.1794