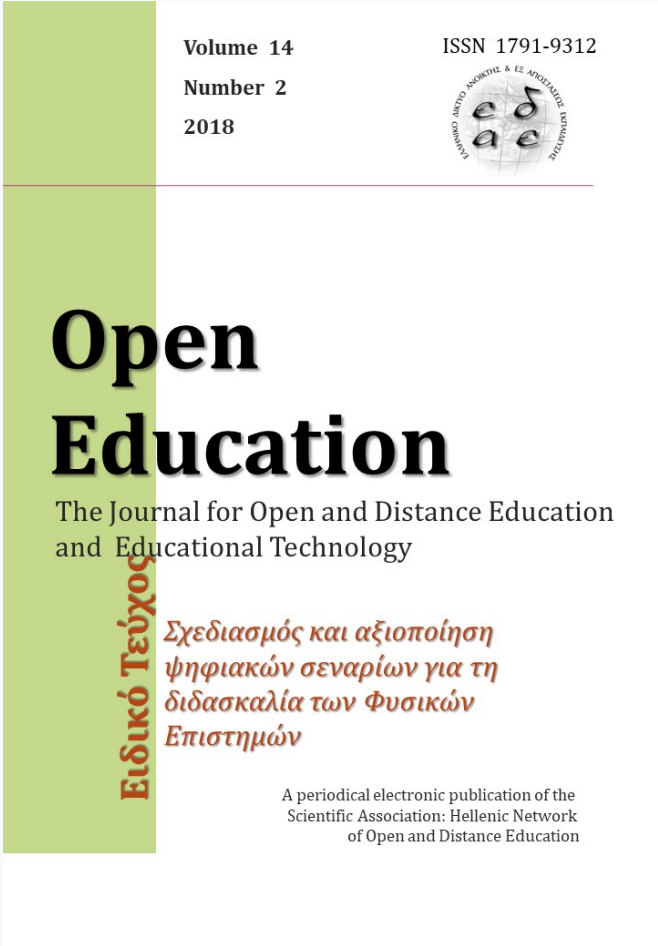
ePublishing
ΕΘΝΙΚΟ ΚΕΝΤΡΟ ΤΕΚΜΗΡΙΩΣΗΣ
NATIONAL DOCUMENTATION CENTRE

Ανοικτή Εκπαίδευση: το περιοδικό για την Ανοικτή και εξ Αποστάσεως Εκπαίδευση και την Εκπαιδευτική Τεχνολογία

Τόμ. 14, Αρ. 2 (2018)

Σχεδιασμός και αξιοποίηση ψηφιακών σεναρίων για τη διδασκαλία των Φυσικών Επιστημών



Αξιοποίηση των φορητών τεχνολογιών και της ρομποτικής για την διδασκαλία του προγραμματισμού σε όλες τις εκπαιδευτικές βαθμίδες

Σταμάτιος Παπαδάκης

doi: 10.12681/jode.19014

Βιβλιογραφική αναφορά:

https://epublishing.ekt.gr | **e-Εκδότης:** EKT | **Πρόσβαση:** 22/07/2025 19:06:06

Αξιοποίηση των φορητών τεχνολογιών και της ρομποτικής για την διδασκαλία του προγραμματισμού σε όλες τις εκπαιδευτικές βαθμίδες

The use of mobile technologies and robotics to teach programming at all educational levels

Σταμάτιος Παπαδάκης
Δρ Επιστημών Αγωγής, Καθηγητής Πληροφορικής
Μεταδιδακτορικός Ερευνητής – Πανεπιστήμιο Κρήτης
stpapadakis@gmail.com
https://orcid.org/0000-0003-3184-1147

Abstract
Although programming is seen as one of the keys to raising standards for the next generation, its teaching and learning at all levels of education presents several problems that are related both to the nature of the lesson and the way it is taught in a classroom. As a consequence of this, students regard programming as one of the most difficult components of the curriculum. In this paper, we present a teaching scenario that attempts to stimulate students' interest in basic programming concepts learning. In this scenario, we combine a tool that students use on a daily basis such as their smart mobile devices with activities that have been proven to attract students interest such as the use of robotics.

Keywords
Smart mobile devices, educational robotics, novice programmers, MIT App Inventor, visual blocked based programming

Περίληψη
Παρότι διεθνώς ο προγραμματισμός θεωρείται ως ο νέος γραμματισμός, η διδασκαλία του σε όλες τις εκπαιδευτικές βαθμίδες παρουσιάζει αρκετά προβλήματα τα οποία σχετίζονται τόσο με την φύση του μαθήματος όσο και με τον τρόπο διδασκαλίας του. Απόρροια τούτου είναι το γεγονός ότι αφενός οι μαθητές θεωρούν τον προγραμματισμό ως ένα δύσκολο και βαρετό μάθημα, αφετέρου όταν τους δίδεται η δυνατότητα είτε δεν επιλέγουν μαθήματα πληροφορικής είτε στην περίπτωση επιλογής τους παρουσιάζουν ιδιαίτερα υψηλά ποσοστά αποχωρήσεων. Στην παρούσα εργασία παρουσιάζεται ένα διδακτικό σενάριο το οποίο προσπαθεί να κεντρίσει το ενδιαφέρον των μαθητών για τον προγραμματισμό και τις συναφείς του έννοιες. Το συγκεκριμένο σενάριο είναι ελεύθερα διαθέσιμο τόσο στην Πλατφόρμα «Αίσωπος» στη διεύθυνση http://aesop.iep.edu.gr/node/11425/2723 όσο και στον εθνικό συσσωρευτή εκπαιδευτικού περιεχομένου με την ονομασία «Φωτόδεντρο» στην διεύθυνση http://photodentro.edu.gr/aggregator/lo/photodentro-aggregatedcontent-8526-8403. Το σενάριο έχει χαρακτηριστεί ως «Βέλτιστο», ύστερα από αξιολόγηση που πραγματοποιήθηκε από δύο ανεξάρτητους αξιολογητές έπειτα από σχετική πρόσκληση του Ινστιτούτου Εκπαιδευτικής Πολιτικής (ΙΕΠ). Το διδακτικό σενάριο διδάσκει στους μαθητές τις βασικές προγραμματιστικές δομές συνδυάζοντας ένα εργαλείο καθημερινής χρήσης των μαθητών-μαθητριών όπως είναι το έξυπνο κινητό

τους τηλέφωνο με δραστηριότητες οι οποίες έχει αποδειχτεί ότι προσελκύουν το ενδιαφέρον τους όπως είναι η αξιοποίηση ρομποτικών κατασκευών.

Λέξεις-κλειδιά
έξυπνες φορητές συσκευές, εκπαιδευτική ρομποτική, αρχάριοι προγραμματιστές, MIT App Inventor, οπτικός προγραμματισμός με πλακίδια

Programming is part of logical thinking and is one of the key skills of the 21st century, like problem-solving, creativity and logical thinking, considered a valuable resource for the human being in the Creativity Society. In this context, the countries of the European Union (EU), apart from their participation in organized activities which aim to boost digital skills, including programming, for different target groups, such as the European Code Week (http://codeweek.eu/), promote the reform of the educational curricula to incorporate - those who have not already done so - the teaching of programming as part of the curriculum activities, either optional or compulsory (European Schoolnet, 2014). For instance, coding has been on the curriculum for primary and secondary school pupils in the UK since September 2014. Additionally, computer programming and coding is already part of the curriculum in 12 countries: Bulgaria, Cyprus, Czech Republic, Denmark, Estonia, Greece, Ireland, Italy, Lithuania, Poland and Portugal (Laplante, 2017). Except for EU, there are also many efforts around the world aimed at introducing children to computer science and programming, from non-profit organization such as code.org, Khan Academy, or the Raspberry Pi Foundation (Broll et al., 2018).

Thus, we would expect that students would warmly embrace programming courses in formal education as the route to their advancement. Evidence suggests, however, that such a trend is not observed. Evidence from the United States and Europe confirms that an increasing number of students opting not to attend courses or lectures relating to programming in both secondary and tertiary education (Papadakis, Kalogiannakis, Orfanakis, & Zaranis, 2014). Another problem is that the number of girls enrolled in technological fields is much less than the number of boys (Cussó-Calabuig, Farran & Bosch-Capblanch, 2018). The difficulty in understanding the abstract programming concepts, as well as the traditional teaching approach that is characterized by lack of interactivity, reduces students' motivation and additionally their interest in learning programming (Piteira and Haddad 2011 as cited in Giannakoulas & Xinogalos, 2018, p. 2). So, it's not a surprise that students consider the teaching and learning of programming to be a hard, painful, and boring learning process (Robins, Rountree, & Rountree, 2003). Educators have been following the classic didactic approach for teaching novices in introductory programming courses for decades. According to Xinogalos and Satratzemi the classic approach to teaching programming is based on:

- a general-purpose programming language that is too big and too idiosyncratic,
- a professional programming environment for that programming language, and
- a set of problems from the area of number and symbol processing (Xinogalos & Satratzemi, 2004, p. 60) that are hostile and alien to them.

In general-purpose programming languages, novice programmers engaged by learning strict syntax rules (see Image 4) rather than to a deeper understanding of programming constructs and abstract concepts (Zapušek & Rugelj, 2013). Thus, novice programmers usually fail to combine abstract concepts and knowledge into

program design, even though they know the syntax and semantics of individual statement (Chiu, 2014).

Over the last few decades, there has been an increasing interest in the development of tools, new pedagogies, and approaches to help students to learn to programme since the days of Logo in 1967. Especially, the last fifteen years the advent of new novice-friendly programming environments that provide robust pathways for transitioning into more complex projects or languages has helped novice programmers to learn in a relatively easy and interesting way the basic concepts of programming. As Blikstein (2018, p. 18) points out research has been showing that child-friendly, block-based graphical programming offers many benefits to young learners when compared to text-based languages. Some of those programming environments include the Scratch, Blockly, Alice and App Inventor. These blocks-based programming environments consist of the use of graphical blocks instead of text commands. Instead of requiring text commands from the novice programmer, this new format supported GUI system allows novices to use a mouse pointer to select commands and put them together similar to a Jigsaw puzzle. As Weintrop & Wilensky (2015, p. 200) note in such environments, learners can assemble functioning programs using only a mouse by snapping together instructions and receiving visual (and sometimes audio) feedback informing the user if a given construction is valid. These increase student interest in this scientific process and successfully lower the barriers faced by novice programmers. Some of the most successful environments are Scratch, Kodu, Logo, Greenfoot, and Alice (Maloney, Resnick, Rusk, Silverman, & Eastmond, 2010; Giannakos, Hubwieser, & Chrisochoides, 2013; Brennan & Resnick, 2012). The latest entry into the world of ILE is the free software called App Inventor for Android (AIA). AIA exploits the increasing popularity of smart mobile devices as well as the trend of users of all ages towards these devices to dynamically penetrate in a variety of in a variety of formal, non-formal and informal learning environments including the home, school, and outdoors (Papadakis et al., 2014).

In this paper, we have shown that the approach of teaching programming using the block-based programming environments AIA and robotics have helped students to significantly improve their performance in terms of basic programming knowledge. Moreover, the results of this study are consistent with previous studies that confirm the superiority of the AIA environment in relation to general-purpose languages for teaching programming to upper secondary school students (Chen & Huang, 2017; Nikou & Economides, 2014). It seems that, as AIA facilitates interactions with the world outside of the classroom through the use of smart mobile devices contribute to retain students' interest and attitudes in programming and thus significantly improve their knowledge of basic programming concepts. This study also confirmed the significant impact of the developmentally appropriate use of smart mobile devices for educational purposes. We must have in mind that the use of AIA facilitates the concept of bringing your own device or BYOD. This model enables students to use their personal mobile devices as a tool that potentially supports teaching and learning.

In this context, we believe that App Inventor can be used as a teaching tool for high school students as it gives teachers more opportunities for their daily interventions to be successful while at the same time give students an opportunity to reflect programming on their daily experiences via their smart mobile devices.

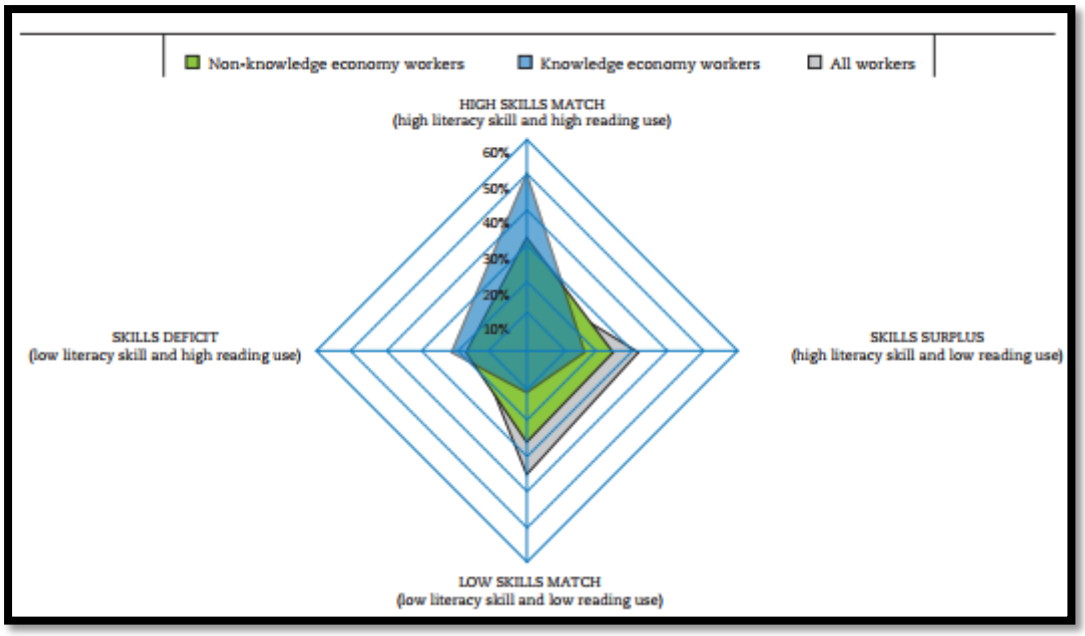
1. Εισαγωγή

Στην παρούσα εργασία παρουσιάζεται ένα ψηφιακό διδακτικό σενάριο με τίτλο «Καθοδήγηση Lego Mindstorms EV3 με τη χρήση του App Inventor». Το

συγκεκριμένο σενάριο είναι ελεύθερα διαθέσιμο τόσο στην Πλατφόρμα «Αίσωπος» στη διεύθυνση http://aesop.iep.edu.gr/node/11425/2723 όσο και στον εθνικό συσσωρευτή εκπαιδευτικού περιεχομένου με την ονομασία «Φωτόδεντρο» στην διεύθυνση http://photodentro.edu.gr/aggregator/lo/photodentro-aggregatedcontent-8526-8403. Το σενάριο έχει χαρακτηριστεί ως «Βέλτιστο», ύστερα από αξιολόγηση που πραγματοποιήθηκε από δύο ανεξάρτητους αξιολογητές έπειτα από σχετική πρόσκληση του Ινστιτούτου Εκπαιδευτικής Πολιτικής (ΙΕΠ), (http://aesop.iep.edu.gr/sites/default/files/kritiria-aksiologisis-senarion.pdf). Παρότι το παρόν διδακτικό σενάριο προτείνεται για χρήση στην Πρωτοβάθμια και στη Δευτεροβάθμια εκπαίδευση στο γνωστικό αντικείμενο της Πληροφορικής λόγω του περιεχομένου που πραγματεύεται μπορεί να αξιοποιηθεί εξίσου στα πλαίσια μιας ολιστικής προσέγγισης της μάθησης. Ως εκ τούτου προτείνεται να χρησιμοποιηθεί στα πλαίσια υλοποίησης του μαθήματος της ερευνητικής εργασίας στην Α' και Β' Λυκείου των Γενικών και Επαγγελματιών Λυκείων αλλά και στα πλαίσια της νέας διαδικασίας υλοποίησης των Δημιουργικών Εργασιών στο Γενικό Λύκειο (Ν4521/18 ΦΕΚ 38Α/2-3-2018)

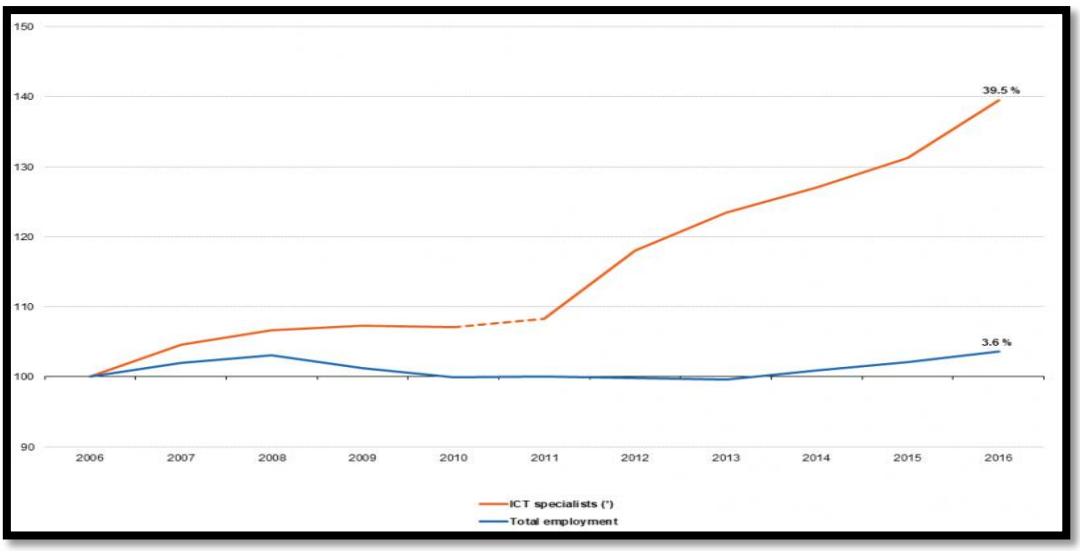
2. Η αναγκαιότητα της ανάπτυξης δεξιοτήτων στον προγραμματισμό

Τα τελευταία χρόνια η ενσωμάτωση της διδασκαλίας του προγραμματισμού στο σχολικό πρόγραμμα έχει αποκτήσει διεθνώς αυξημένη βαρύτητα. Ένας βασικός λόγος είναι ότι ο προγραμματισμός στη σημερινή Κοινωνία της Πληροφορίας, εκτός από ένα μαθησιακό αντικείμενο, έχει μετατραπεί σε μια βασική δεξιότητα που απευθύνεται σε όλους τους μαθητές, καθώς είναι ένα εργαλείο για κάθε μελλοντικό εργαζόμενο. Όπως έχει επισημάνει ο Οργανισμός Οικονομικής Συνεργασίας και Ανάπτυξης (ΟΟΣΑ) ήδη από το 2013, οι απαιτήσεις για συγκεκριμένες δεξιότητες στην σημερινή κοινωνία της γνώσης υποδεικνύουν ότι τα άτομα με ανεπαρκείς δεξιότητες του 21ου αιώνα (21st century skills) είναι πιο πιθανό να βρεθούν σε κίνδυνο ανεργίας και κοινωνικού αποκλεισμού (βλέπε Εικόνα 1). Οι δεξιότητες του 21ου αιώνα αφορούν ένα ευρύ φάσμα δεξιοτήτων όπως δεξιότητες μάθησης και καινοτομίας (δηλαδή, κριτική σκέψη, δημιουργικότητα, συνεργασία και επικοινωνία) και δεξιότητες πληροφόρησης, μέσων και τεχνολογίας (Qian & Clark, 2016, π. 51).
Επιπρόσθετα, η Ένωση Καθηγητών Πληροφορικής στις ΗΠΑ (CSTA) η οποία αριθμεί πάνω από 16.000 μέλη σε ολόκληρο τον κόσμο και αποτελεί θυγατρικό οργανισμό της ένωσης Association for Computer Machinery (ACM), διευρύνει το επιχείρημα για την ανάπτυξη της κωδικοποίησης και της υπολογιστικής σκέψης (computational thinking) επισημαίνοντας την αξία των δυο προαναφερθέντων τομέων-εννοιών για την ενίσχυση ενός ανωτέρου επιπέδου σκέψης των μαθητών και των δυνατοτήτων επίλυσης προβλημάτων (Falloon, 2016, σελ. 580). Όπως χαρακτηριστικά επισημαίνουν οι Chen & Huang (2017) ο προγραμματισμός είναι ένας εξαιρετικός τρόπος για την ανάπτυξη των δεξιοτήτων υπολογιστικής σκέψης των μαθητών, καθώς περιλαμβάνει τη χρήση εννοιών της επιστήμης των υπολογιστών, όπως η αφαίρεση, η αποσφαλμάτωση και η επανάληψη για την επίλυση προβλημάτων (Psycharis,2018; Psycharis & Kalia, 2017).



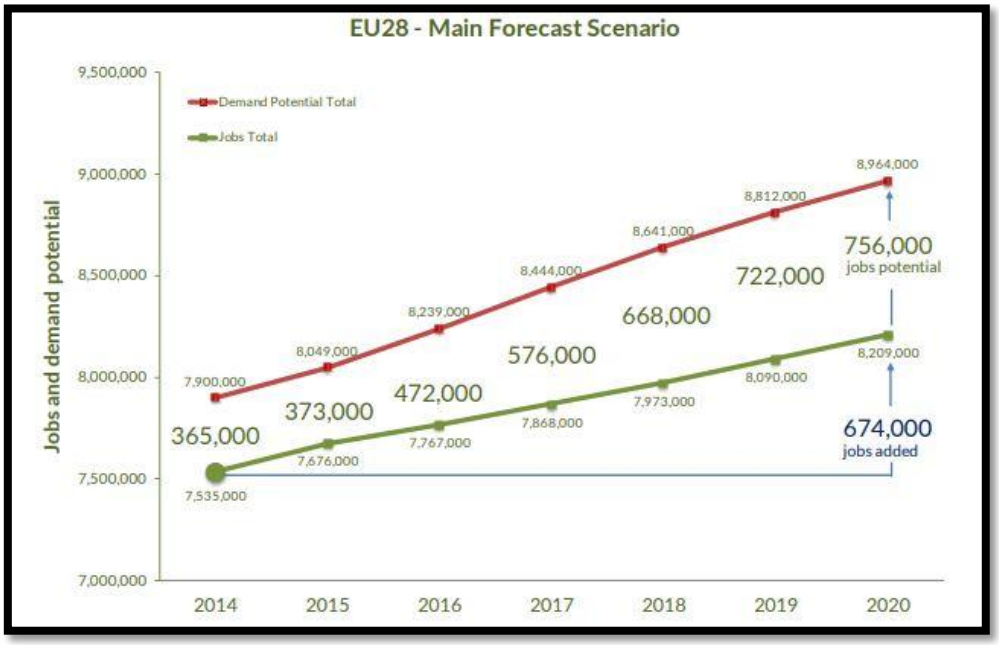
Εικόνα 1: Καταγραφή δεξιοτήτων από τον ΟΟΣΑ (Πηγή: OECD, 2013).

Διεθνώς ο τομέας των Τεχνολογιών της Πληροφορίας και των Επικοινωνιών (ΤΠΕ – ICT) χαρακτηρίζεται ως ο χώρος με τις περισσότερες δυνατότητες επαγγελματικής ανάπτυξης, δεδομένου ότι δημιουργεί οικονομική ανάπτυξη και νέες θέσεις εργασίας, έχοντας την δυναμική να οδηγήσει σε ποικίλες τεχνολογικές εξελίξεις (European Schoolnet, 2014). Σύμφωνα με το Βρετανικό οργανισμό SFIA (Skills Framework for the Information Age), υπάρχουν 86 διαφορετικές επαγγελματικές δεξιότητες που σχετίζονται με την Πληροφορική και αντιστοιχούν σε 290 διαφορετικές ειδικότητες που απαιτούνται προκειμένου να μπορέσουν οι σημερινές εταιρείες να «πλοηγηθούν» στο σύνθετο σύμπαν της τεχνολογίας των πληροφοριών (SFIA Foundation, 2017). Ορισμένοι από αυτούς τους τομείς σχετίζονται με τον προγραμματισμό εφαρμογών για έξυπνες φορητές συσκευές, τη δημιουργία και συντήρηση δικτυακών τόπων, τη αξιοποίηση εφαρμογών υπηρεσιών νέφους, κ.α. Σύμφωνα με την Eurostat, ο αριθμός των απασχολούμενων στον τομέα των ΤΠΕ στην Ευρωπαϊκή Ένωση (ΕΕ) των 28 αυξήθηκε κατά 39,5% κατά την περίοδο 2006 έως 2016, ποσοστιαία άνοδος η οποία ήταν περισσότερο από 10 φορές υψηλότερη από την αντίστοιχη αύξηση (3,6%) για τη συνολική απασχόληση (βλέπε Εικόνα 2). Ταυτόχρονα, σε όλη την ΕΕ των 28, η μεγάλη πλειονότητα των θέσεων εργασίας στις ΤΠΕ το 2016 κυριαρχήθηκαν από άνδρες εργαζόμενους. Το μερίδιο των γυναικών εργαζομένων στις ΤΠΕ ήταν 16,7%, περίπου 5,8 ποσοστιαίες μονάδες λιγότερο σε σύγκριση με το αντίστοιχο ποσοστό μια δεκαετία πριν (Eurostat, 2018). Η χαμηλή παρουσία των γυναικών σε θέσεις εργασίας που συνδέονται με τις ΤΠΕ έχει βρεθεί ότι σχετίζεται με την χαμηλή συμμετοχή των γυναικών ήδη από τα μαθητικά χρόνια σε μαθήματα προγραμματισμού και σε συναφείς δραστηριότητες στις σχολικές μονάδες (Papadakis, 2018b; Papadakis, Tousia & Polychronaki, 2018) με επακόλουθο αποτέλεσμα την χαμηλή συμμετοχή των γυναικών και σε πιο εξειδικευμένες σπουδές στον τεχνολογικό τομέα (Cussó-Calabuig, Farran & Bosch-Capblanch, 2018).



Εικόνα 2. Απασχολούμενοι στις ΤΠΕ και σύνολο απασχόλησης, EU-28, 2006-2016 (Πηγή: Eurostat, 2018 url: https://goo.gl/Z46udK)

Στις 10 Ιουνίου 2016, η Ευρωπαϊκή Επιτροπή δημοσίευσε μια νέα ατζέντα δεξιοτήτων για την Ευρώπη, υπογραμμίζοντας το γεγονός ότι η εξέλιξη της τεχνολογίας έχει τεράστιο αντίκτυπο στην αγορά εργασίας καθώς προσδιορίζει το είδος των δεξιοτήτων που απαιτούνται στην οικονομία και την κοινωνία. Η Ευρωπαϊκή Επιτροπή εκτιμά ότι μέχρι το 2020 θα υπάρχουν 500.000 κενές θέσεις εργασίας για προσωπικό το οποίο σχετίζεται με τον τομέα των ΤΠΕ (European Commission, 2017). Ωστόσο, οι εκτιμήσεις αυτές πιθανότατα αποδεικνύονται συντηρητικές, καθώς άλλες εκθέσεις προβλέπουν ότι η ζήτηση για δεξιότητες στις ΤΠΕ θα συνεχίσει να αυξάνεται με πιο γοργούς ρυθμούς. Το 2020 προβλέπεται ότι η ευρωπαϊκή αγορά εργασίας θα χρειαστεί περισσότερες από 670.000 νέες θέσεις εργασίας στις ΤΠΕ, αλλά θα μπορέσει να απορροφήσει επιπλέον 756.000 απασχολούμενους στις ΤΠΕ, εφόσον υπάρχει επαρκής προσφορά στο αντίστοιχο εργατικό δυναμικό (Hüsing, Korte & Dashja, 2015) (βλέπε Εικόνα 3).



Εικόνα 3. Σενάριο πρόβλεψης: Απασχόληση και ζήτηση στον τομέα των ΤΠΕ στην Ευρώπη (EU-27) 2014 – 2020 (Πηγή: Hüsing, Korte, & Dashja, 2015).

3. Η δυσκολία της διδασκαλίας του προγραμματισμού

Η διδασκαλία των μαθημάτων προγραμματισμού στην Πρωτοβάθμια και Δευτεροβάθμια Εκπαίδευση αποσκοπεί στην απόκτηση γνώσεων για τις προγραμματιστικές έννοιες/δομές, στην απόκτηση ικανοτήτων, στην σχεδίαση/υλοποίηση λύσεων και στην επίλυση προβλημάτων χρησιμοποιώντας ποικίλα εργαλεία (Γρηγοριάδου, Γόγουλου, & Γουλή, 2002; Papadakis, 2018a). Ωστόσο, η εκμάθηση του προγραμματισμού από τους μαθητές της Δευτεροβάθμιας εκπαίδευσης είναι ένα δύσκολο εγχείρημα, καθώς είναι μια εγγενώς πολύπλοκη διανοητική δραστηριότητα (Εφόπουλος, Δαγδιλέλης & Ευαγγελίδης, 2005; Papadakis & Orfanakis, 2017; 2018; Psycharis & Kalia, 2017). Σύμφωνα με τον Blikstein (2018, σελ. 32), ο προγραμματισμός απαιτεί από τους μαθητές να:

- καταλάβουν πως λειτουργούν οι υπολογιστές και πώς εκτελούν προγράμματα,
- ερμηνεύουν, να εντοπίζουν και να διορθώνουν λάθη στον παραγόμενο κώδικα,
- προσπαθήσουν να υπερνικήσουν ποικίλες παρερμηνείες οι οποίες συχνά σχετίζονται με την διδασκαλία λοιπών γνωστικών αντικειμένων (π.χ. Μαθηματικά),
- διαχειριστούν το γνωστικό φορτίο που απαιτεί η διδασκαλία του προγραμματισμού,
- κατανοήσουν τους συμβολισμούς και τις συμβάσεις που απαιτούν οι ποικίλες γλώσσες προγραμματισμού,
- γνωρίζουν το περιεχόμενο από άλλους επιστημονικούς κλάδους (π.χ. ανάγνωση, αριθμητική, άλγεβρα, μεταβλητές) και να κατανοούν τις αλληλεπικαλύψεις και τις αντιφάσεις τους με τον προγραμματισμό και,
- εργάζονται σε μέσης ή μακράς διάρκειας έργα (projects) σε περιβάλλοντα και μεθόδους οι οποίοι διαφέρουν ριζικά από το παραδοσιακό περιβάλλον της σχολικής τάξης.

Ως εκ τούτου, δεν αποτελεί έκπληξη το γεγονός ότι οι μαθητές θεωρούν τη διδασκαλία και εκμάθηση του προγραμματισμού μια σκληρή, επίπονη και βαρετή διαδικασία μάθησης (Robins, Rountree, & Rountree, 2003). Επιπλέον, ο Guzdial (2017 όπως αναφέρεται στο Blikstein, 2018, σελ. 9) επισημαίνει ότι, γενικά, οι μαθητές κατά τη διδασκαλία του προγραμματισμού αποτυγχάνουν να εφαρμόσουν ακόμη και απλές ιδέες σε αρκετά απλά προβλήματα. Η βιβλιογραφία σχετιζόμενη με την διδασκαλία του προγραμματισμού είναι γεμάτη με καταγραφή των γνωστικών – νοητικών εμποδίων τα οποία δυσχεραίνουν τους αρχαρίους προγραμματιστές-μαθητές από την εκμάθηση των βασικών δομών - αρχών του προγραμματισμού (Broll et al., 2018; Papadakis, Kalogiannakis, Orfanakis, & Zaranis, 2014; 2016; 2017). Σήμερα, μετά την ερευνητική προσπάθεια 30 και πλέον ετών, επιχειρώντας να καταγράψουμε το πλέγμα δυσκολιών που αντιμετωπίζουν οι μαθητές – αρχάριοι προγραμματιστές στις βασικές προγραμματιστικές έννοιες και δομές, καταλήγουμε στο συμπέρασμα ότι οι μαθητές αντιμετωπίζουν προβλήματα που σχετίζονται με:

- την διδασκαλία και κατανόηση των μεταβλητών,
- την διδασκαλία και κατανόηση των επαναληπτικών δομών (βρόχων),
- την διδασκαλία και κατανόηση των εντολών επιλογής,
- την διδασκαλία της αναδρομικότητας,
- ειδικά θέματα, όπως την μεταφορά γνώσεων από ένα προγραμματιστικό περιβάλλον σε άλλο, την εισαγωγή στον λογικό προγραμματισμό και τον παράλληλο προγραμματισμό κ.ά. (Du Boulay, 1989; Εφόπουλος, 2005; Τζιμογιάννης, 2015).

Επιπρόσθετα, τα παραπάνω προβλήματα δυσχεραίνονται και από την παραδοσιακή διδακτική προσέγγιση η οποία ακολουθείται στην διδασκαλία των αρχαρίων προγραμματιστών, καθώς χαρακτηρίζεται από έλλειψη διαδραστικότητας η οποία ακολούθως μειώνει τα κίνητρα των μαθητών και αναπόφευκτα το ενδιαφέρον τους για την εκμάθηση του προγραμματισμού (Piteira και Haddad 2011 όπως αναφέρεται στον Giannakoulas & Xinogalos, 2018 σελ. 2). Η κλασική διδακτική προσέγγιση του προγραμματισμού βασίζεται στα εξής:

- μια γλώσσα προγραμματισμού γενικού σκοπού που είναι αρκετά σύνθετη και ιδιοσυγκρασιακή,
- ένα επαγγελματικό περιβάλλον προγραμματισμού για τη συγκεκριμένη γλώσσα προγραμματισμού και,
- ένα σύνολο προβλημάτων κυρίως από την περιοχή επεξεργασίας αριθμών και συμβόλων τα οποία είναι «εχθρικά» και ξένα προς τους μαθητές (Xinogalos & Satratzemi, 2004, σελ. 60).

Στις γλώσσες προγραμματισμού γενικού σκοπού, οι αρχάριοι προγραμματιστές ασχολούνται με την εκμάθηση αυστηρών κανόνων σύνταξης (Orfanakis & Papadakis, 2014; 2016) (βλέπε Εικόνα 4) και όχι με μια βαθύτερη κατανόηση των δομών προγραμματισμού και των αφηρημένων εννοιών (Zapušek & Rugelj, 2013). Έτσι, οι αρχάριοι προγραμματιστές συνήθως αποτυγχάνουν να συνδυάσουν αφηρημένες έννοιες και γνώσεις στο σχεδιασμό του προγράμματος, παρόλο που γνωρίζουν τη σύνταξη και τη σημασιολογία των μεμονωμένων δηλώσεων (Chiu, 2014). Για παράδειγμα, ο Guzdial (2018) αναφέρει ότι τα προγράμματα ηλεκτρονικών υπολογιστών θα μπορούσαν να οδηγήσουν σε παρερμηνείες για τους μαθητές λόγω λανθασμένης μεταφοράς υπάρχουσας γνώσης από τα Μαθηματικά. Για παράδειγμα αναφέρει ότι στην γλώσσα προγραμματισμού Python η χρήση του συμβόλου "=" χρησιμοποιείται με διαφορετικό τρόπο σε σχέση με τα Μαθηματικά και ως εκ τούτου η χρήση του συμβόλου αυτού ("=") δημιουργεί ποικίλες παρενέργειες κατά τη σύνταξη ενός προγράμματος από τους αρχάριους προγραμματιστές.

```
public class HelloWorld {

    public static void main(String[] args) {

        System.out.println("Hello, World");
    }

}
```

(α)

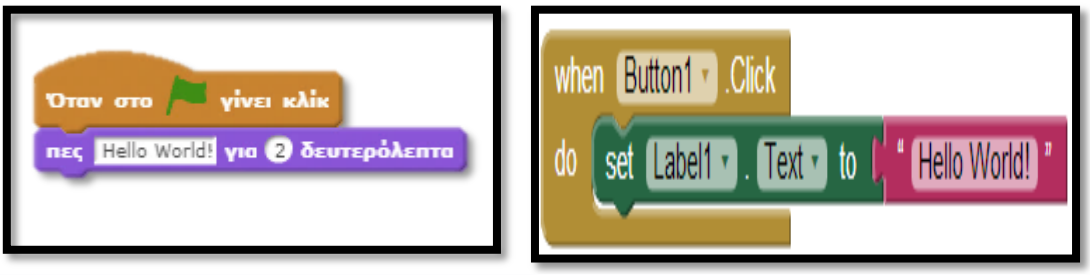
```
#include <iostream>
using namespace std;

int main()
{
    cout << "Hello, World!";
    return 0;
}
```

(β)

Εικόνα 4 (α, β). Παραδείγματα χρήσης κώδικα για την εμφάνιση του μηνύματος “Hello, World!” με την χρήση δυο από τις πιο δημοφιλείς κειμενικές γλώσσες προγραμματισμού, την Java και την C++.

Για τους προαναφερθέντες λόγους, οι εμπλεκόμενοι με τη διδασκαλία του προγραμματισμού αναζητούν διαρκώς νέες προσεγγίσεις και διδακτικές στρατηγικές οι οποίες θα είναι περισσότερο αποδοτικές και φιλικές για τους αρχαρίους προγραμματιστές (Papadakis & Orfanakis, 2017). Για το σκοπό αυτό, εξακολουθούν να αναζητούν διδακτικά μοντέλα χρησιμοποιώντας ποικίλες γλώσσες προγραμματισμού και περιβάλλοντα, εργαλεία, τεχνικές, κατάλληλα και αυθεντικά παραδείγματα καθώς και αλλαγές στο περιεχόμενο της διδασκαλίας (Kelleher & Pausch, 2005). Μια από τις πιο επιτυχημένες προσπάθειες διδασκαλίας του προγραμματισμού στους αρχάριους είναι η ανάπτυξη και η χρήση εξειδικευμένων

περιβαλλόντων προγραμματισμού προσανατολισμένα για αρχάριους προγραμματιστές (Guzdial, 2004). Όπως τονίζει ο Guzdial (2004, σελ. 127), «*κάθε περιβάλλον προγραμματισμού αρχαρίων επιχειρεί να απαντήσει στην ερώτηση, τι κάνει τον προγραμματισμό δύσκολο;*». Τα περιβάλλοντα προγραμματισμού για αρχάριους ή περιβάλλοντα αρχικής εκμάθησης (Initial Learning Environments - ILE) (Fincher, Cooper, Kölling & Maloney, 2010) είναι εκπαιδευτικά περιβάλλοντα τα οποία είναι οπτικά, αποσκοπούν στην άμεση συμμετοχή των μαθητών – αρχαρίων σε μια ελκυστική δραστηριότητα μάθησης και επιτρέπουν στους αρχάριους προγραμματιστές ανεξαρτήτως ηλικίας, φύλου και εκπαιδευτικού υποβάθρου τη δημιουργία διαδραστικών εφαρμογών, έργων, και εφαρμογών μέσω μιας ειδικά διαμορφωμένης διεπαφής χρήστη (Graphical User Interface, GUI). Παρέχουν δε στους αρχάριους προγραμματιστές ένα εργαλείο ανάπτυξης το οποίο τους επιτρέπει να δημιουργούν εφαρμογές ενώνοντας απλά κομμάτια εντολών (πλακίδια) με τον ίδιο τρόπο με τον οποίο ενώνουν κομμάτια παζλ (Papadakis et al., 2014; 2016; 2017) (βλέπε Εικόνα 5). Ως εκ τούτου, τα περιβάλλοντα προγραμματισμού που βασίζονται σε πλακίδια αυξάνονται σε δημοτικότητα και χρησιμοποιούνται ολοένα και περισσότερο σε εισαγωγικά μαθήματα προγραμματισμού (Weintrop, 2015).

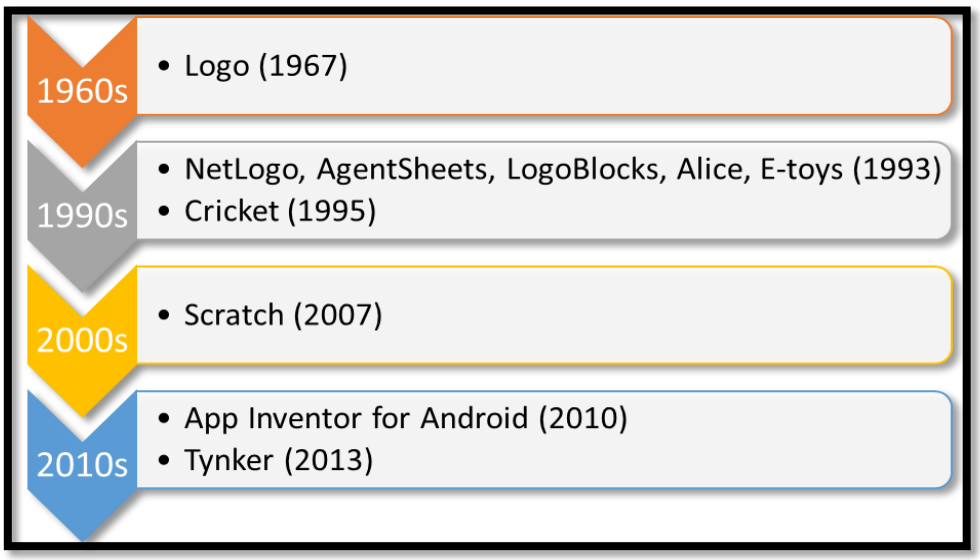


(α) **(β)**

Εικόνα 5 (α, β). Παραδείγματα χρήσης κώδικα για την εμφάνιση του μηνύματος "Hello, World!" με την χρήση δυο από τις πιο δημοφιλείς οπτικές γλώσσες προγραμματισμού, το Scratch και το App Inventor

Τα περιβάλλοντα προγραμματισμού που βασίζονται σε χρήση πλακιδίων συνίστανται στη χρήση γραφικών δομικών στοιχείων εντολών αντί για εντολές κειμένου. Το μεγαλύτερο πλεονέκτημα το οποίο απορρέει από την χρήση τους είναι ότι αντί να απαιτούν την σύνταξη εντολών κειμένου από τον αρχάριο προγραμματιστή (συχνά με δυσνόητο και δύσχρηστο συντακτικό), επιτρέπουν στους αρχάριους να χρησιμοποιούν τον δείκτη του ποντικιού για να επιλέγουν εντολές και να τις τοποθετούν τις περισσότερες φορές την μια πάνω από την άλλη (Papadakis, 2018a). Όπως σημειώνουν οι Weintrop & Wilensky (2015, σελ. 200) σε τέτοια περιβάλλοντα, οι εκπαιδευόμενοι μπορούν να δημιουργήσουν προγράμματα χρησιμοποιώντας μόνο το ποντίκι τους, συνδυάζοντας τα πλακίδια εντολών και λαμβάνοντας άμεσα οπτική (και μερικές φορές και ακουστική) ανατροφοδότηση σχετικά με την εγκυρότητα των ενεργειών τους. Τα συγκεκριμένα χαρακτηριστικά αφενός αυξάνουν το ενδιαφέρον των αρχαρίων για τον προγραμματισμό και αφετέρου μειώνουν με σχετική επιτυχία τα γνωστικά εμπόδια – νοητικό φόρτο που αντιμετωπίζουν. Μερικά από τα πιο επιτυχημένα προγραμματιστικά περιβάλλοντα για αρχαρίους είναι το Scratch, το Kodu, η Logo, το Greenfoot και η Alice. Η τελευταία είσοδος στον κόσμο των ILE είναι το MIT App Inventor for Android (AIA). Το AIA εκμεταλλεύεται την αυξανόμενη δημοτικότητα των έξυπνων φορητών συσκευών καθώς και την τάση των χρηστών όλων των ηλικιών προς αυτές τις συσκευές προκειμένου να «διεισδύσει»

δυναμικά σε μια ποικιλία τυπικών, μη τυπικών και άτυπων μαθησιακών περιβαλλόντων (Papadakis & Orfanakis, 2018) (βλέπε Εικόνα 6).



Εικόνα 6. Χρονοδιάγραμμα εμφάνισης περιβαλλόντων προγραμματισμού για αρχαρίους (Προσαρμογή από Blikstein, 2018, pp. 38-39)

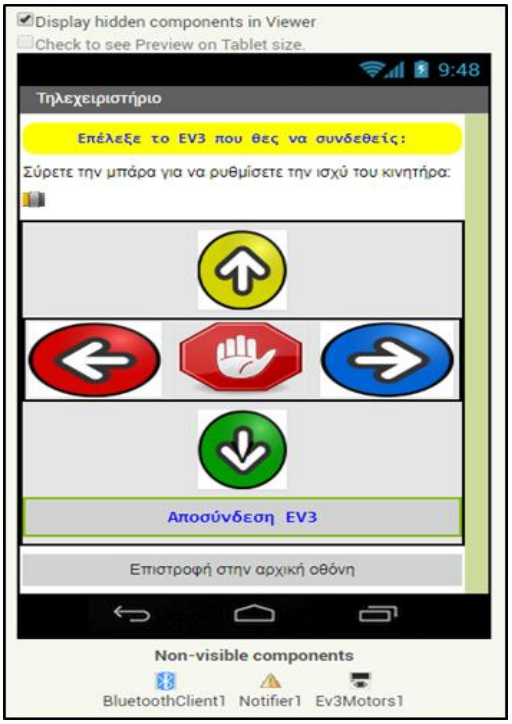
4. Παρουσίαση του διδακτικού σεναρίου

4.1 Γενική περιγραφή

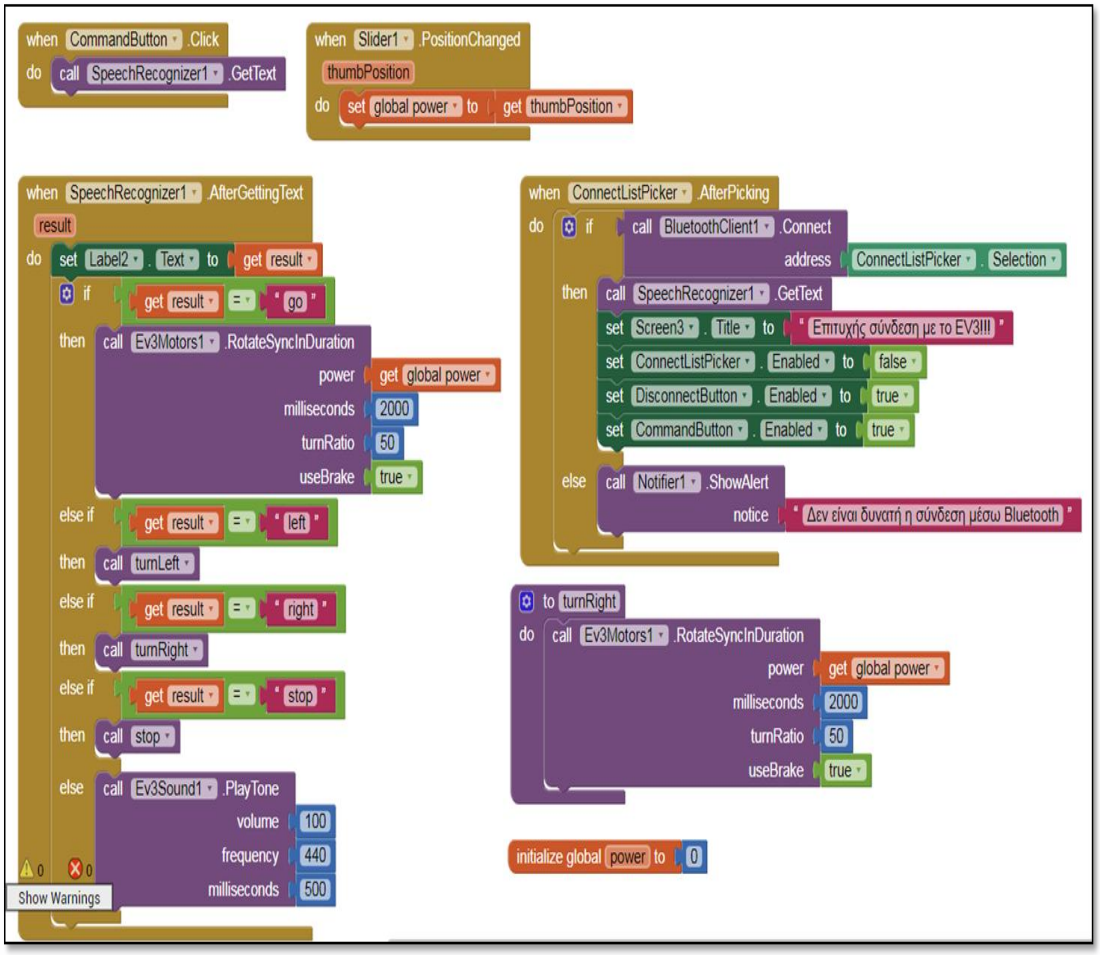
Το διδακτικό Σενάριο με τίτλο «Καθοδήγηση Lego Mindstorms EV3 με τη χρήση του App Inventor» έχει δημιουργηθεί με σκοπό να χρησιμοποιηθεί ως μια επαναληπτική δραστηριότητα για την εμπέδωση των βασικών προγραμματιστικών δομών. Στα πλαίσια του σεναρίου οι μαθητές καλούνται να δημιουργήσουν μια φορητή εφαρμογή η οποία καθοδηγεί ένα όχημα – ρομπότ τύπου Lego Mindstorms EV3. Η καθοδήγηση μπορεί να λάβει χώρα με δυο τρόπους: είτε μέσω της χρήσης φωνητικής καθοδήγησης είτε μέσω της χρήσης τηλεχειριστηρίου. Μέσω της υλοποίησης του σεναρίου οι μαθητές εισάγονται με έναν ιδιαίτερα διαδραστικό τρόπο στις βασικές αρχές του δομημένου προγραμματισμού όσο και σε έννοιες του αντικειμενοστραφούς προγραμματισμού όπως: του αντικειμένου (object), της ιδιότητας (property) και της μεθόδου (method). Είναι προφανές ότι το παρόν σενάριο μπορεί να υλοποιηθεί εναλλακτικά με την παλαιότερη έκδοση Lego Mindstorms NXT με τις αντίστοιχες προσαρμογές στην χρήση των συστατικών Lego Mindstorms του App Inventor.

Ταυτόχρονα ενώ οι μαθητές μέσω του σεναρίου καλούνται να επιλύσουν με τρόπο δημιουργικό ένα πρόβλημα που τους κινεί το ενδιαφέρον και το οποίο σχετίζεται με ένα αντικείμενο καθημερινής χρήσης όπως το κινητό τους τηλέφωνο, το σενάριο είναι άμεσα εμπλεκόμενο με θέματα της διδασκόμενης ύλης σχετικά με το μάθημα της Πληροφορικής, εμβαθύνοντας τις γνώσεις τους γύρω από το συγκεκριμένο ζήτημα ή/και αναπλαισιώνοντάς τις με τρόπο κριτικό, πρωτότυπο, και ευρηματικό. Ταυτόχρονα η απαίτηση του σεναρίου για την σχεδίαση του γραφικού περιβάλλοντος διεπαφής προσφέρει στους μαθητές και μαθήτριες να εκφραστούν δημιουργικά, σχεδιάζοντας και υλοποιώντας μία ψηφιακή κατασκευή (βλέπε Εικόνες 7, 8 και 9) σε ένα μαθησιακό περιβάλλον το οποίο ευνοεί τη διαφοροποιημένη διδακτική προσέγγιση.

Εικόνα 7. Παράδειγμα γραφικού περιβάλλοντος χρήστη της φορητής εφαρμογής του σεναρίου



Εικόνα 8. Παράδειγμα σχεδίασης γραφικού περιβάλλοντος της φορητής εφαρμογής



Εικόνα 9. Απόσπασμα κώδικα της φορητής εφαρμογής

4.2 Ένταξη του διδακτικού σεναρίου στο πρόγραμμα σπουδών

Το παρόν διδακτικό σενάριο μπορεί να χρησιμοποιηθεί στην Πρωτοβάθμια και στη Δευτεροβάθμια εκπαίδευση. Στη Δευτεροβάθμια εκπαίδευση μπορεί να διδαχθεί τόσο στην υποχρεωτική (Γυμνάσιο) όσο και στην μεταϋποχρεωτική εκπαίδευση η οποία προσφέρεται από το Γενικό και το Επαγγελματικό Λύκειο. Στο πρόγραμμα Σπουδών του μαθήματος «Εφαρμογές Πληροφορικής» της Α΄ τάξης του Γενικού Λυκείου (ΦΕΚ Β" 932 / 14-04-2014) αναφέρεται ότι, το μάθημα έχει σκοπό οι μαθητές να μπορούν να αναλύουν προβλήματα, να σχεδιάζουν και να αναπτύσσουν μικροεφαρμογές των ηλεκτρονικών υπολογιστών, «έξυπνων» κινητών συσκευών, κ.α. Ειδικότερα στη θεματική ενότητα με τίτλο «Προγραμματιστικά περιβάλλοντα - Δημιουργία Εφαρμογών» (διάρκειας 12 ωρών) αναφέρεται η υλοποίηση ή τροποποίηση μικροεφαρμογής με χρήση διαφόρων δωρεάν προγραμματιστικών εργαλείων συμπεριλαμβανομένων του App Inventor. Στην Α' Λυκείου του ΕΠΑΛ (ΦΕΚ 2211/2013 - Αριθμ.116605/Γ2 αναφέρεται ότι το μάθημα Εφαρμογές Πληροφορικής (διδασκαλίας 2 ή 4 ωρών εβδομαδιαίως) έχει ως γενικό σκοπό: την επέκταση της γενικής πληροφορικής παιδείας των μαθητών με έμφαση στην ανάπτυξη ικανοτήτων και δεξιοτήτων στη χρήση και αξιοποίηση των υπολογιστικών και δικτυακών τεχνολογιών ως εργαλείων μάθησης, σκέψης, έκφρασης επικοινωνίας και συνεργασίας. Ειδικότερα το συγκεκριμένο σενάριο θα μπορούσε να χρησιμοποιηθεί στη Θεματική ενότητα με τίτλο «Λογισμικό Εφαρμογών - Προγραμματισμός» διάρκειας 30 ή 16 ωρών αντίστοιχα.

Στο Γυμνάσιο και στις 3 τάξεις (Ομάδα Εργασίας, 2011) υπάρχει η ενότητα «Προγραμματίζω τον υπολογιστή μου». Βασικός στόχος της ενότητας αυτής είναι η σταδιακή εξοικείωση των μαθητών με τον προγραμματισμό μέσα από την αξιοποίηση διαθέσιμων εκπαιδευτικών περιβαλλόντων οπτικού προγραμματισμού. Ειδικότερα αναφέρεται ότι με την ολοκλήρωση της ενότητας ο μαθητής και μαθήτρια εκτός των άλλων θα πρέπει να είναι ικανός και ικανή να αντιλαμβάνεται την αναγκαιότητα και τη χρησιμότητα των δομών επανάληψης και επιλογής να χρησιμοποιεί εντολές επανάληψης και επιλογής στα προγράμματα που αναπτύσσει.
Στο Δημοτικό στα πλαίσια του μαθήματος Τεχνολογίες Πληροφορίας και Επικοινωνιών της ΣΤ' Δημοτικού στην ενότητα «Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ» διάρκειας 38 ωρών υπάρχει και ο άξονας «Προγραμματίζω τον υπολογιστή» διάρκειας 12 ωρών. Στις προβλεπόμενες δραστηριότητες περιλαμβάνονται η υλοποίηση κατάλληλων δραστηριοτήτων ώστε να επεκτείνουν και να ενισχύσουν τις προγραμματιστικές τους δεξιότητες. Επίσης στην ενότητα με τίτλο «Υλοποιώ σχέδια έρευνας με ΤΠΕ» διάρκειας 8-10 ωρών υπάρχει η υποενότητα εκπαιδευτική ρομποτική.
Σύμφωνα με τις υπάρχουσες οδηγίες οι μαθητές σε ομάδες των 3-4 ατόμων σχεδιάζουν και οργανώνουν την εργασία τους, διακρίνουν τα μέσα και τα εργαλεία του περιβάλλοντος της εκπαιδευτικής ρομποτικής. Συναρμολογούν το ρομπότ, σχεδιάζουν, υλοποιούν, ελέγχουν και βελτιώνουν απλούς και σύνθετους αλγόριθμους καθοδήγησης του ρομπότ. Ενδεικτικά παραδείγματα είναι ο υπολογισμός περιμέτρου και εμβαδού επιφάνειας, η μέτρηση απόστασης με βάση την περίμετρο της ρόδας του ρομπότ, η κίνηση του ρομπότ σε λαβύρινθο ή χώρο με εμπόδια, με ανταπόκριση σε ηχητικές εντολές.
Τέλος λόγω φύσης του σεναρίου και της διαθεματικής προσέγγισης της ύλης που δυνητικά μπορεί να πραγματευτεί, προτείνεται να χρησιμοποιηθεί στα πλαίσια υλοποίησης του μαθήματος της ερευνητικής εργασίας στις τάξεις Α και Β Λυκείου των Γενικών και Επαγγελματιών Λυκείων αλλά και στα πλαίσια της νέας διαδικασίας υλοποίησης των Δημιουργικών Εργασιών στο Γενικό Λύκειο (Ν4521/18 ΦΕΚ 38Α/2-3-2018).

4.3 Χρήση προγραμματιστικού περιβάλλοντος – ρομποτικής κατασκευής

Το λογισμικό MIT App Inventor for Android (AIA) (εφευρέτης εφαρμογών) παρότι δεν αναφέρεται στα πρόσφατα προγράμματα σπουδών του Δημοτικού και του Γυμνασίου, παρά μόνο στο πρόγραμμα σπουδών του μαθήματος «Εφαρμογές Πληροφορικής» της Α΄ τάξης του Γενικού Λυκείου κρίνουμε ότι μπορεί με ιδιαίτερη επιτυχία να χρησιμοποιηθεί σε όλες τις εκπαιδευτικές βαθμίδες.
Ο λόγος είναι ότι το AIA είναι ένα ιδιαίτερα εύχρηστο αλλά και «ισχυρό» προγραμματιστικό περιβάλλον που μπορεί να ανταποκριθεί και στις πιο απαιτητικές μαθησιακές δραστηριότητες (Papadakis et al., 2014; 2016; 2017). Το περιβάλλον είναι αρκετά πρόσφατο, καθώς δημιουργήθηκε ως ένα πειραματικό έργο στα εργαστήρια της εταιρείας Google (Google Labs), μόλις το 2009. Ωστόσο, σε αντίθεση με άλλα προγραμματιστικά περιβάλλοντα για αρχαρίους όπως το Alice ή στο Scratch, στο AIA, οι μαθητές δε δημιουργούν εφαρμογές οι οποίες εκτελούνται στους συμβατικούς υπολογιστές αλλά σε έξυπνες φορητές συσκευές με λειτουργικό σύστημα Android (smartphones / tablets). Πορίσματα ερευνών δείχνουν πως οι μαθητές έχουν αυξημένα κίνητρα για τη χρήση νέων τεχνολογιών και ότι προσεκτικά σχεδιασμένες παιδαγωγικές δραστηριότητες με τη χρήση των έξυπνων φορητών συσκευών παρακινούν τους μαθητές να εμπλακούν με αυτές (Bradley & Holley, 2011). Ουσιαστικά, το AIA εκμεταλλεύεται τη ραγδαία απήχηση που έχουν οι

φορητές συσκευές στον νεανικό πληθυσμό προκειμένου να κεντρίσει το ενδιαφέρον των μαθητών στον προγραμματισμό (Orfanakis & Papadakis, 2016).

Η εκπαιδευτική ρομποτική εξασφαλίζει ένα νέο τρόπο προσέγγισης της Πληροφορικής που μπορεί να εξάψει το ενδιαφέρον των μικρών μαθητών, καθώς έρχονται σε επαφή με σημαντικές έννοιες της επιστήμης των υπολογιστών παρέχοντας τα κίνητρα για μάθηση βασισμένη στην κατασκευή αντικειμένων που έχουν νόημα για τους ίδιους (Ατματζίδου, Μαρκέλης & Δημητριάδης, 2008). Βασικό εργαλείο της είναι το προγραμματιζόμενο ρομπότ, οντότητα προικισμένη με αυτονομία, ικανή να εκπληρώσει συγκεκριμένες εκ των προτέρων ενέργειες μέσα σε ένα μεταβαλλόμενο περιβάλλον (Φράγκου & Παπανικολάου, 2010). Οι Καγκάνη, Δαγδιλέλης, Σατρατζέμη & Ευαγγελίδης (2005) σχολιάζουν ότι η προσέγγιση της διδασκαλίας του προγραμματισμού με τα Lego Mindstorms, ενδεχόμενα να συμβάλει στην εξάλειψη των αδυναμιών που συνεπάγεται η παραδοσιακή μέθοδος και να δημιουργήσει τις κατάλληλες συνθήκες μάθησης, ώστε να γίνει αποτελεσματικότερη η διδασκαλία.

Τα Lego Mindstorms πρωτοεμφανίστηκαν το 1998 και ανήκουν στην κατηγορία των λεγόμενων «kit 3ης γενιάς» (http://www.LegoMindstorms.com). Πρόκειται για ένα σύνθετο προϊόν που συνδυάζει προγραμματιζόμενα «τούβλα» με ηλεκτρικές μηχανές, αισθητήρες, απλά τούβλα, και τεχνικά κομμάτια όπως άξονες, ακτίνες, και υδραυλικά μέρη, κατάλληλα για να χτίσει ο χρήστης διάφορα ρομπότ και άλλα αυτοματοποιημένα συστήματα. Οι Φράγκου & Παπανικολάου (2010) αναφέρουν ότι ως εκπαιδευτικά εργαλεία τα εκπαιδευτικά ρομπότ της εταιρίας Lego έχουν παρομοιαστεί με τα χαρακτηριστικά ενός δωματίου που έχει «χαμηλό δάπεδο, ψηλό ταβάνι και είναι ευρύχωρο» («low floor, high ceiling and wide walls»). Είναι δηλαδή εργαλεία τα οποία εύκολα γίνονται προσιτά σε αρχάριους, είναι όμως εμπλουτισμένα με πολλές δυνατότητες τις οποίες μπορεί να χρησιμοποιήσει και ένας ειδικός, αλλά κυρίως είναι κατάλληλα για την υλοποίηση ποικίλων ιδεών (Resnick & Silverman, 2005 όπ. αναφ. στο Φράγκου & Παπανικολάου, 2010).

Εφαρμογές της εκπαιδευτικής ρομποτικής με μαθητές και εκπαιδευτικούς έδειξαν θετικά αποτελέσματα στην ανάπτυξη του τεχνολογικού εγγραμματισμού και της επίλυσης προβλημάτων. Η αξιοποίηση της ρομποτικής σε συνδυασμό με τον οπτικό προγραμματισμό δημιουργεί ένα εντελώς νέο περιβάλλον εργασίας για τους μαθητές. Ειδικότερα, επιτυγχάνεται υψηλός βαθμός αλληλεπίδρασης μεταξύ υπολογιστή και πραγματικού αντικειμένου, υπάρχει άμεση ανατροφοδότηση, ο εκπαιδευτικός διαθέτει περισσότερο χρόνο για κάθε μαθητή, υλοποιείται ένα είδος εξατομικευμένης μάθησης, υπάρχει πειραματισμός και ενεργός συμμετοχή από τους μαθητές, αναπτύσσεται η κριτική σκέψη, ενώ ο μαθητής απαλλάσσεται από την εκμάθηση και απομνημόνευση συντακτικών κανόνων μιας γλώσσας προγραμματισμού (Χαρίσης & Μικρόπουλος, 2008).

4.4 Σκοπός του διδακτικού σεναρίου

Τα φτωχά μαθησιακά αποτελέσματα από την εισαγωγή του προγραμματισμού στις σχολικές μονάδες εγείρουν αμφιβολίες για τις μεθόδους που χρησιμοποιήθηκαν για αυτή την εισαγωγή (Forte & Guzdial, 2004). Ιδιαίτερα, η απουσία κινήτρου αποτελεί ένα από τους κύριους λόγους για τους οποίους οι μαθητές εγκαταλείπουν πρόωρα μαθήματα προγραμματισμού (Siegle, 2009). Οι μαθητές περιγράφουν τα μαθήματα προγραμματισμού ως υπερβολικά τεχνικά, αποκομμένα από τον πραγματικό κόσμο και στερούμενα δημιουργικότητας (Khuloud & Gestwicki, 2013). Οι Forte και Guzdial (2004) θεωρούν ότι η κύρια αιτία αποτυχίας ή πρόωρης εγκατάλειψης των μαθημάτων προγραμματισμού αποτελεί η αντίληψη των μαθητών ότι δεν είναι

ενδιαφέροντα ή χρήσιμα. Η διδασκαλία του προγραμματισμού θα πρέπει να γίνεται με τρόπο, ώστε να ελαχιστοποιείται το γνωστικό φορτίο, ενώ ταυτόχρονα να μεγιστοποιείται η παιδαγωγική αξία. Η εμπλοκή των μαθητών είναι συχνά επιτυχής, όταν το πλαίσιο διδασκαλίας τροφοδοτείται από θέματα που έχουν άμεσο ενδιαφέρον για τους μαθητές (Gray, Abelson, Wolber, & Friend, 2012).

Το παρόν διδακτικό σενάριο λαμβάνοντας υπόψη τα προαναφερθέντα προσπαθεί να μετατρέψει τον εισαγωγικό προγραμματισμό σε μια εύκολη και διασκεδαστική εμπειρία για τους αρχάριους προγραμματιστές. Τον σκοπό αυτό προσπαθεί να τον επιτύχει με ποικίλους τρόπους. Ένας τρόπος είναι μέσω της μείωσης του ενδογενούς γνωστικού φορτίου που απαιτείται από τους αρχάριους για την εκμάθηση του προγραμματισμού, με αντίστοιχη μείωση της ποσότητας των πληροφοριών που χρησιμοποιούνται για την επίλυση ενός προβλήματος. Τα φύλλα εργασίας που παρέχονται στους μαθητές και μαθήτριες είναι δομημένα με τέτοιο τρόπο ώστε οι αρχάριοι προγραμματιστές να διδαχθούν την επίλυση ενός προβλήματος πρώτα σε θεωρητικό επίπεδο, δημιουργώντας αρχικά τα απαραίτητα νοητικά μοντέλα, τα οποία επικεντρώνονται στην κατασκευή των λύσεων, δίχως οι ίδιοι να ασχοληθούν ιδιαίτερα με τη σύνταξη δυσνόητων εντολών. Η επιλογή του AIA ως περιβάλλοντος ανάπτυξη εφαρμογών εξυπηρετεί τον σκοπό αυτό αφού οι γλώσσες τύπου «σύρε και άσε» (drag-and-drop) αντικαθιστούν τον προς σύνταξη κώδικα με συρόμενα συστατικά μέρη (components), προσέγγιση η οποία μειώνει το γνωστικό φορτίο, το σχετιζόμενο με τη σύνταξη των εντολών, επιτρέποντας στους χρήστες να επικεντρωθούν στην εννοιολογική επίλυση ενός προβλήματος.

Ταυτόχρονα η έρευνα έχει δείξει ότι όταν οι μαθητές ασχολούνται με μαθησιακά αντικείμενα τα οποία είναι εγγύτερα στα ενδιαφέροντά τους συμμετέχουν με περισσότερο ενθουσιασμό στα μαθήματα προγραμματισμού και εν τέλει σημειώνουν καλύτερα μαθησιακά αποτελέσματα. Στα πλαίσια αυτά η επιλογή του προγραμματιστικού περιβάλλοντος AIA παρέχει επιπλέον κίνητρα στους μαθητές σε σχέση με το Scratch και το Alice εξαιτίας της φορητότητας και της πρακτικής χρήσης των εφαρμογών που δημιουργούνται, καθώς οι μαθητές έχουν τη δυνατότητα να δουν άμεσα τις εφαρμογές να τρέχουν στις έξυπνες κινητές τους συσκευές. Επιπλέον, η ύπαρξη emulator (προσομοιωτή) προσδίδει ευελιξία στη διδασκαλία του AIA, αφού δεν καθιστά υποχρεωτική την ύπαρξη έξυπνων φορητών συσκευών εντός του σχολικού εργαστηρίου πληροφορικής, εξαλείφοντας προβλήματα που σχετίζονται με την αγορά και συντήρηση εξοπλισμού αλλά και διδακτικού θορύβου και εφαρμογής της σχετικής νομοθεσίας.

Επιπρόσθετα η χρήση των ρομπότ για την εισαγωγή σε θέματα προγραμματισμού εκτιμάται ότι μπορεί να είναι θετική, αφού μπορεί να βοηθήσει- μεταξύ άλλων- στην κατανόηση μιας ακριβούς και λογικής γλώσσας εντολών (Κόμης, 2004). Τα ρομπότ μπορεί να χρησιμοποιηθούν ως ένα μέσο διδασκαλίας μεθόδων επίλυσης προβλημάτων, αποτελώντας μία ευχάριστη και ενδιαφέρουσα ενασχόληση παρέχοντας παράλληλα, μία απλή και διδακτική διεπαφή. Οι μαθητές τα αντιμετωπίζουν περισσότερο ως παιχνίδι, παρά ως εργαλεία μάθησης καθώς η πλειοψηφία τους έχει «παίξει» με αυτά. Η πτυχή-παιχνίδι, αποτελεί ένα σημαντικό παράγοντα θετικού κινήτρου και παρότρυνσης στην εκπαίδευση (Κόμης, 2005; Papadakis, 2018c; Papadakis & Kalogiannakis, 2017).

4.5 Πλαίσιο υλοποίησης του διδακτικού σεναρίου

Για την υλοποίηση του διδακτικού σεναρίου οι μαθητές εργάζονται ατομικά ή σε ομάδες 2-3 μαθητών (ανάλογα την διαθεσιμότητα του εργαστηρίου Πληροφορικής). Ο εκπαιδευτικός μοιράζει πολλαπλά φύλλα εργασίας. Οι μαθητές μέσω των φύλλων

εργασίας και των δραστηριοτήτων τους προσπαθούν να κατανοήσουν τις έννοιες του αντικειμενοστραφούς και δομημένου προγραμματισμού στο προγραμματιστικό περιβάλλον ΑΙΑ. Σε όλα τα στάδια της διδακτικής δραστηριότητας ο εκπαιδευτικός δεν περιορίζεται στο απλό μοίρασμα δραστηριοτήτων και στην παθητική παρακολούθηση των μαθητών αλλά συμμετέχει ενεργά στη μαθησιακή διαδικασία, παρακολουθώντας και ελέγχοντας τις απαντήσεις τους, συμβουλεύοντας τους κριτικά κατά τη φάση επίλυσης των ασκήσεων ή ανάπτυξης των αλγορίθμων ενθαρρύνοντας τους να συνεχίσουν την διερευνητική τους προσπάθεια μέσω στοχευόμενων ερωτήσεων και προβληματισμών. Συνοψίζοντας η βοήθεια που προσφέρει ο εκπαιδευτικός στους μαθητές, άλλοτε ρητά και άλλοτε άρρητα, μπορεί να είναι είτε υποστηρικτική, συνερευνητική ή και καθοδηγητική έχοντας υπόψη του ότι οι μαθητές μαθαίνουν αλληλεπιδρώντας με το περιβάλλον τους και ότι τα προβλήματα αποτελούν το κριτήριο και την πηγή της γνώσης.

Αναγκαίο είναι ο εκπαιδευτικός να έχει ετοιμάσει τα προγράμματα σε διάφορες ανεξάρτητες εκδοχές τους (π.χ. το σχεδιαστικό κομμάτι καθώς και το ολοκληρωμένο πρόγραμμα) και να έχει φροντίσει ιδίως τα προγράμματα με το σχεδιαστικό τμήμα να τα έχει ήδη τοποθετήσει σε ένα δημόσιο αποθετήριο προσβάσιμο από τους μαθητές είτε τοπικά στους σταθμούς εργασίας του εργαστηρίου Πληροφορικής.

Στο παρόν σενάριο δίδεται βαρύτητα στο προγραμματιστικό κομμάτι και όχι στο κατασκευαστικό. Ως εκ τούτου ο εκπαιδευτικός στο πλαίσιο της εφαρμογής των δραστηριοτήτων, θα πρέπει να επικεντρώσει την προσοχή του στην εκμάθηση ή/και των αρχών του προγραμματισμού και όχι στην ανάπτυξη κατασκευαστικών ικανοτήτων. Για το λόγο αυτό, το φυσικό μοντέλο Lego (το αυτοκίνητο) προτείνεται να δοθεί έτοιμο στους μαθητές ώστε να προχωρήσουν κατευθείαν στον προγραμματισμό της συμπεριφοράς του.

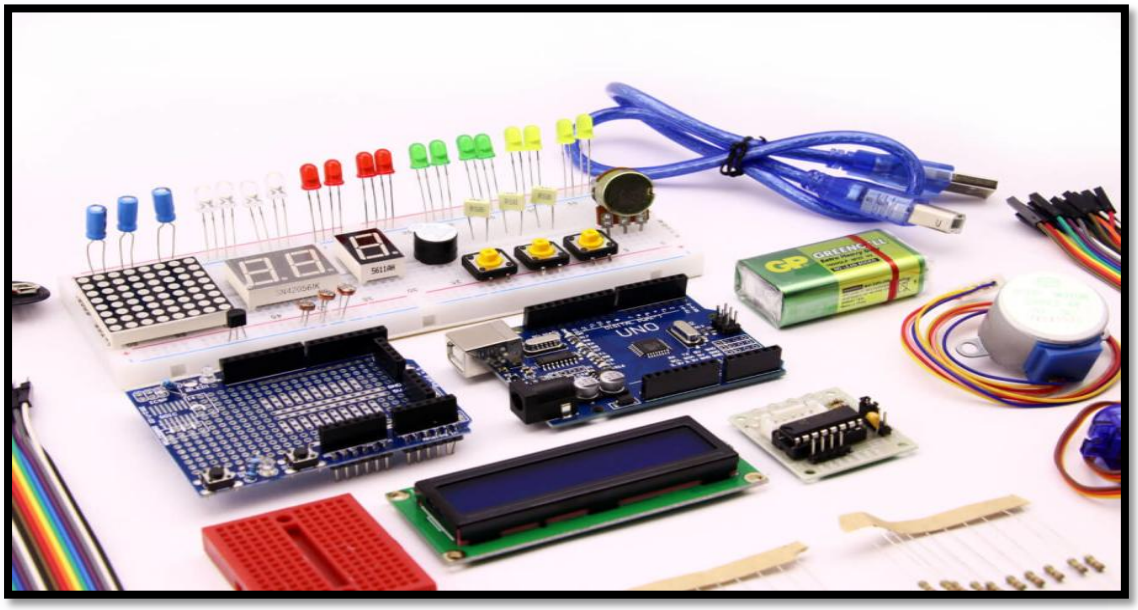
4.6 Παιδαγωγικό πλαίσιο σεναρίου

Το σενάριο είναι θεμελιωμένο στις βασικές ιδέες της εποικοδομικής (constructivist) αντίληψης για τη μάθηση (Piaget, 1972) και την «κατασκευαστική» εκπαιδευτική φιλοσοφία (constructionism) του Papert σύμφωνα με τις οποίες η κατασκευή νέας γνώσης είναι περισσότερο αποτελεσματική όταν οι μαθητές εμπλέκονται στην κατασκευή προϊόντων που έχουν προσωπικό νόημα για τους ίδιους (Papert, 1993). Στο συγκεκριμένο διδακτικό σενάριο ο μαθητής χτίζει την γνώση του ανιχνεύοντας, διερευνώντας και αλληλοεπιδρώντας οπτικά με τις βασικές εντολές της περιβάλλοντος προγραμματισμού του ΑΙΑ δημιουργώντας μια ολοκληρωμένη εφαρμογή καθοδήγησης ενός ρομποτικού οχήματος.

4.7 Επεκτάσεις / διασυνδέσεις των εννοιών ή των δραστηριοτήτων

Μετά την ολοκλήρωση του σεναρίου οι μαθητές και μαθήτριες θα είναι προετοιμασμένοι για να εργαστούν ατομικά ή ομαδικά στο τυπικό ή σε άτυπα περιβάλλοντα μάθησης με δραστηριότητες οι οποίες θα περιλαμβάνουν την ανάπτυξη ολοκληρωμένων φορητών εφαρμογών. Επίσης στα πλαίσια της διαθεματικής προσέγγισης της ύλης, το συγκεκριμένο σενάριο θα μπορούσε να χρησιμοποιηθεί με τα γνωστικά αντικείμενα του STEM (Science - Επιστήμη, Technology - Τεχνολογία, Engineering - Μηχανική, Mathematics - Μαθηματικά) καθώς σχετίζεται στα διάφορα στάδια υλοποίησης του με την διδασκαλία των Μαθηματικών και των Φυσικών Επιστημών (Psycharis,2018). Το περιβάλλον ΑΙΑ μέσω της διαρκούς εξέλιξης του, προσφέρει πλέον στους χρήστες του μια παλέτα από έτοιμες υλοποιήσεις (συστατικά μέρη - components) προκειμένου να αξιοποιήσουν στο έπακρο τα χαρακτηριστικά του ρομποτικού οχήματος αλλά και δημιουργήσουν εφαρμογές που ποικίλα διδακτικά

αντικείμενα όπως Φυσική, Μαθηματικά κ.α. Ενδεικτικά αναφέρονται οι πιθανές χρήσεις των αισθητήρων υπερήχων, γυροσκοπίου κ.α.
Για την αποφυγή του αυξημένου κόστους απόκτησης του ρομποτικού πακέτου (κιτ) Lego Mindstorms EV3, εναλλακτικά θα μπορούσε να χρησιμοποιηθεί ένας μικροεπεξεργαστής τύπου Arduino ή και Raspberry Pi. Το ιδιαίτερα χαμηλό κόστος των δυο μικροεπεξεργαστών όσο και των αισθητήρων και των υλικών συνδεσμολογίας επιτρέπει την παροχή ολόκληρου σετ σε κάθε ομάδα, χωρίς το δυσβάσταχτο οικονομικό κόστος της λύσης τύπου Lego (βλέπε Εικόνα 10). Ακόμη και αν προς το παρόν το περιβάλλον AIA δεν επιτρέπει άμεσα την αλληλεπίδραση με τους συγκεκριμένους μικροεπεξεργαστές, η ενασχόληση των μαθητών και μαθητριών με το περιβάλλον του AIA θα τους δώσει την δυνατότητα να μεταπηδήσουν σε συναφή περιβάλλοντα προγραμματισμού όπως το Scratch for Arduino (S4A) (http://s4a.cat/).



Εικόνα 10. Κιτ για το Arduino

4.8 Ανάπτυξη του σεναρίου

Για την υλοποίηση του διδακτικού σεναρίου έχουν δημιουργηθεί 2 φύλλα εργασίας κάθε ένα από τα οποία αποτελείται από 3 επιμέρους δραστηριότητες. Επιπρόσθετα έχει δημιουργηθεί και ένα φύλλο εργασίας για την επανάληψη – εμπέδωση των γνώσεων από τους μαθητές αλλά κυρίως για να χρησιμοποιηθεί ως ένα εργαλείο ανατροφοδότησης από την πλευρά του εκπαιδευτικού.

5. Συζήτηση

Οι μαθητές ενώ δείχνουν μεγάλο ενδιαφέρον για το διαδίκτυο, για λογισμικά γενικής χρήσης και, κυρίως, για τα ηλεκτρονικά παιχνίδια, δεν ισχύει το ίδιο και για τον προγραμματισμό. Αρκετές έρευνες δείχνουν ότι ο προγραμματισμός αποτελεί, για την πλειονότητα των μαθητών, μια ελάχιστα ελκυστική δραστηριότητα (Ελευθεριώτη, Καρατράντου & Παναγιωτακόπουλος, 2010). Οι μαθητές αντιλαμβάνονται τον προγραμματισμό ως μια δύσκολη, κουραστική, βαρετή, άνευ σημασίας και αντικοινωνική εργασία. Ως αποτέλεσμα, μειώνεται συνεχώς ο αριθμός των μαθητών που μαθαίνουν προγραμματισμό τα τελευταία χρόνια. Οι Forte & Guzdial (2004)

υποστηρίζουν ότι η «παραδοσιακή» διδακτική προσέγγιση στον προγραμματισμό είναι πιθανότερο να αποτρέψει, παρά να προσελκύσει τους μαθητές.

Στα πλαίσια της διδασκαλίας των μαθημάτων Πληροφορικής, απαιτούνται κατάλληλες διδακτικές καταστάσεις και περιβάλλοντα που θα βοηθούν τους μαθητές να επιλύουν προβλήματα, αφού συγκρουστούν με τις παρανοήσεις τους, προκειμένου να δημιουργήσουν στέρεες και λειτουργικές αναπαραστάσεις για την κατανόηση εννοιών και δομών προγραμματισμού (Τζιμογιάννης, Πολίτης & Κόμης, 2005). Η διδασκαλία των μαθημάτων πρέπει να γίνεται με τέτοιο τρόπο, ώστε να παρέχει στους μαθητές ένα ενδιαφέρον και πλούσιο σε ερεθίσματα περιβάλλον, μέσα από το οποίο προβλήματα και έννοιες γίνονται αντικείμενο διερεύνησης με δημιουργικό και ευχάριστο τρόπο. Με τη συνδυασμένη χρήση των εργαλείων της ρομποτικής και μιας γλώσσας οπτικού προγραμματισμού (με την οποία ο μαθητής προγραμματίζει μέσω ενός περιβάλλοντος με ενέργειες τύπου «drag & drop»), τα προσδοκώμενα οφέλη για τους εκπαιδευτικούς είναι ο εμπλουτισμός των διδακτικών επιλογών τους με δύο εργαλεία τα οποία εύκολα γίνονται προσιτά σε αρχάριους, είναι όμως εμπλουτισμένα με πολλές δυνατότητες τις οποίες μπορεί να χρησιμοποιήσει και ένας ειδικός, αλλά κυρίως είναι κατάλληλα για την υλοποίηση ποικίλων ιδεών (Resnick & Silverman, 2005).

Οι παραπάνω διαπιστώσεις αποτέλεσαν το κίνητρο για την αναζήτηση νέων μεθόδων διδασκαλίας για τα εισαγωγικά μαθήματα προγραμματισμού, με σκοπό να εξαλειφθούν τα προβλήματα που παρουσιάζει η παραδοσιακή μέθοδος (Καγκάνη κ.α., 2005). Μια διαφορετική προσέγγιση είναι αυτή που αξιοποιεί την εκπαιδευτική ρομποτική σε συνδυασμό με τον οπτικό προγραμματισμό. Η διδασκαλία της ρομποτικής είναι κατάλληλη για μαθητές ανεξάρτητα από την ηλικία και το υπόβαθρό τους και αποτελεί έναν τρόπο ενθάρρυνσης της μάθησης (Χαρίσης & Μικρόπουλος, 2008). Η ρομποτική μπορεί να χρησιμοποιηθεί ως ένα μέσο διδασκαλίας μεθόδων επίλυσης προβλημάτων, αποτελώντας μία ευχάριστη και ενδιαφέρουσα ενασχόληση παρέχοντας παράλληλα, μία απλή και διδακτική διεπαφή. Οι μαθητές αντιμετωπίζουν τη ρομποτική περισσότερο ως παιχνίδι, παρά ως εργαλείο μάθησης ενώ η πλειοψηφία τους έχει «παίξει» ήδη με κάποιο ρομπότ (Ατματζίδου, Μαρκέλης, & Δημητριάδης, 2008).

Στην παρούσα εργασία παρουσιάζεται ένα διδακτικό σενάριο το οποίο αξιοποιεί το πακέτου ρομποτικών κατασκευών Lego Mindstorms EV3 και το περιβάλλον προγραμματισμού με πλακίδια App Inventor. Σκοπός του σεναρίου είναι η δημιουργία φορητής εφαρμογής για συσκευές τύπου Android για το χειρισμό του ρομπότ. Υιοθετώντας το παιχνίδι ως έναυσμα μάθησης το παρόν σενάριο στοχεύει στη δημιουργία θετικών κινήτρων και παρακίνησης των μαθητών προκειμένου να οδηγηθούν στην ευκολότερη κατανόηση των βασικών προγραμματιστικών δομών. Η συνδυασμένη αξιοποίηση των τεχνολογικών εργαλείων για την ανάπτυξη εκπαιδευτικών δραστηριοτήτων στα πλαίσια μαθησιακών και συνεργατικών δραστηριοτήτων επίλυσης προβλήματος, θεωρούμε ότι θα συμβάλει τόσο στην οικοδόμηση των απαραίτητων νοητικών πλαισίων για να εξασκήσουν οι μαθητές και οι μαθήτριες προγραμματιστικές δραστηριότητες, όσο και να μετατρέψει τον προγραμματισμό στα μάτια τους ως μια εύκολη και ελκυστική δραστηριότητα.

Βιβλιογραφία

Blikstein, P. (2018). *Pre-College Computer Science Education: A Survey of the Field*. Mountain View, CA: Google LLC. Ανακτήθηκε στις 15 Μαρτίου, 2018 από: https://goo.gl/gmS1Vm.

Bradley, C., & Holley, D. (2011). Empirical research into students' mobile phones and their use for learning. *International Journal of Mobile and Blended Learning, 3*(4), 38-53.

Broll, B., Lédeczi, Á., Zare, H., Do, D. N., Sallai, J., Völgyesi, P., ... & Vanags, C. (2018). A visual programming environment for introducing distributed computing to secondary education. *Journal of Parallel and Distributed Computing*.

Chen, P., & Huang, R. (2017). Design thinking in App inventor game design and development: A case study. In *Advanced Learning Technologies (ICALT), 2017 IEEE 17th International Conference on* (pp. 139-141). IEEE.

Chiu, C. F. (2014). Use of problem-solving approach to teach scratch programming for adult novice programmers. In Proceedings of the 45th ACM technical symposium on Computer science education (pp. 710-711). ACM.

Cussó-Calabuig, R., Farran, X. C., & Bosch-Capblanch, X. (2018). Effects of intensive use of computers in secondary school on gender differences in attitudes towards ICT: A systematic review. *Education and Information Technologies*, 1-29.

Du Boulay. (1989). Some difficulties of learning to program. In E. Soloway & J.C. Spohrer (Eds.), *Studying the novice programmer* (pp. 283–299). Hillsdale, NJ: Lawrence Erlbaum.

European Commission (2017). *Digital Skills & Jobs*. Ανακτήθηκε στις 16 Μαρτίου, 2018 από: https://ec.europa.eu/digital-single-market/en/policies/digital-skills.

Eurostat. (2018). Persons employed as ICT specialists and total employment, EU-28, 2006-2016. Ανακτήθηκε στις 25 Μαρτίου, 2018 από: http://ec.europa.eu/eurostat/statistics-explained/index.php/File:Persons_employed_as_ICT_specialists_and_total_employment,_EU-28,_2006-2016_ICT17.png.

Falloon, G. (2016). An analysis of young students' thinking when completing basic coding tasks using Scratch Jnr. On the iPad. *Journal of Computer Assisted Learning*, *32*(6), 576-593.

Fincher, S., Cooper, S., Kölling, M., & Maloney, J. (2010). Comparing alice, greenfoot & scratch. In *Proceedings of the 41st ACM technical symposium on Computer science education* (pp. 192-193). ACM.

Forte, A., & Guzdial, M. (2004). Computers for communication, not calculation: Media as a motivation and context for learning. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on* (pp. 10-pp). IEEE.

Giannakoulas, A., & Xinogalos, S. (2018). A pilot study on the effectiveness and acceptance of an educational game for teaching programming concepts to primary school students. *Education and Information Technologies*, 1-24.

Gray, J., Abelson, H., Wolber, D., & Friend, M. (2012). Teaching CS principles with app inventor. In *Proceedings of the 50th Annual Southeast Regional Conference* (pp. 405-406). ACM.

Guzdial, M. (2004). Programming environments for novices. Computer science education research, 2004, 127-154.

Guzdial, M. (2018). *Confusion over the forms of programming problems: Mathematics/Physics versus CS*. Ανακτήθηκε στις 5 Μαρτίου, 2018 από: https://computinged.wordpress.com/2018/02/26/confusion-over-the-forms-of-programming-problems-mathematics-physics-versus-cs/.

Hüsing, T., Korte, W., & Dashja, E. (2015). E-Skills in Europe. Trends and Forecasts for the European ICT Professional and Digital Leadership Labour Markets (2015-2020). empirica Gesellschaft für Kommunikations-und Technologieforschung mbH, Bonn, Germany, 42. Ανακτήθηκε στις 22 Μαρτίου, 2018 από: http://eskills-lead.eu/fileadmin/lead/brochure-lead/working_paper_-_supply_demand_forecast_2015_a.pdf.

Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys (CSUR)*, *37*(2), 83-137.

Khuloud, A., & Gestwicki, P. (2013). Studio-based learning and app inventor for android in an introductory CS course for non-majors. In Proceeding of the 44th ACM technical symposium on Computer science education (SIGCSE '13) (pp. 287-292). New York: ACM.

OECD (2013). *OECD Skills Outlook 2013*. OECD Publishing, Paris. Ανακτήθηκε στις 15 Μαρτίου, 2018 από: *http://dx.doi.org/10.1787/9789264204256-en*.

Orfanakis V. & Papadakis, St. (2014). A new programming environment for teaching programming. A first acquaintance with Enchanting. The 2nd international virtual Scientific Conference - Scieconf 2014 (pp. 268-273). EDIS - University of Zilina, Slovakia.

Orfanakis V. & Papadakis, St. (2016). Teaching basic programming concepts to novice programmers in Secondary Education using Twitter, Python, Ardruino and a coffee machine. In *Proceeding of the Hellenic Conference on Innovating STEM Education (HISTEM)*, 16-18 December 2016 University of Athens, Greece.

Papadakis S., & Orfanakis V. (2017). The Combined Use of Lego Mindstorms NXT and App Inventor for Teaching Novice Programmers. In: *Alimisis D., Moro M., Menegatti E. (Eds.), Educational Robotics in the Makers Era. Edurobotics 2016. Advances in Intelligent Systems and Computing*, vol 560, pp.193-204. Springer, Cham.

Papadakis, S. (2018a). Is Pair Programming More Effective than Solo Programming for Secondary Education Novice Programmers?: A Case Study. *International Journal of Web-Based Learning and Teaching Technologies*, *13*(1), DOI: 10.4018/IJWLTT.2018010101.

Papadakis, S. (2018b). Gender stereotypes in Greek computer science school textbooks. *International Journal of Teaching and Case Studies*, *9*(1), 48-71.

Papadakis, S. (2018c). The use of computer games in classroom environment. *International Journal of Teaching and Case Studies*, *9*(1), 1-25.

Papadakis, S., & Orfanakis, V. (2018). Comparing novice programing environments for use in secondary education: App Inventor for Android vs. Alice. *International Journal of Technology Enhanced Learning*, *10*(1-2), 44-72.

Papadakis, S., Kalogiannakis, M., Orfanakis, V. & Zaranis, N. (2016). Using Scratch and App Inventor for teaching introductory programming in Secondary Education. A case study. *International Journal of Technology Enhanced Learning*, *8*(3/4), 217 – 233.

Papadakis, S., Kalogiannakis, M., Orfanakis, V., & Zaranis, N. (2017). The appropriateness of scratch and app inventor as educational environments for teaching introductory programming in primary and secondary education. *International Journal of Web-Based Learning and Teaching Technologies (IJWLTT)*, *12*(4), 58-77.

Papadakis, S., Tousia, C. & Polychronaki, K. (2018). Women in computer science. The case study of the Computer Science Department of the University of Crete, Greece. *Int. J. Teaching and Case Studies*, *9*(2), 142–151.

Papadakis, St. & Kalogiannakis, M. (2017). Using Gamification for Supporting an Introductory Programming Course. The Case of ClassCraft in a Secondary Education Classroom. *In proceedings of the 2nd EAI International Conference on Design, Learning & Innovation*, October 30–31, 2017 | Heraklion, Greece.

Papadakis, St., Kalogiannakis, M., Orfanakis, V., Zaranis, N. (2014). Novice Programming Environments. Scratch & App Inventor: a first comparison. In *H. M. Fardoun and J. A. Gallud (Eds.) Proceedings of the 2014 Workshop on Interaction Design in Educational Environments* (pp 1-7), New York: ACM.

Psycharis, S., & Kalia, M. (2017). The Effects of Computer Programming on high school students' problem solving, reasoning skills and self-efficacy in Mathematics. *Instructional Science*, *45*(5), 583-602.

Psycharis, S. (2018). STEAM in Education: A Literature review on the role of Computational Thinking, Engineering Epistemology and Computational Science. Computational STEAM Pedagogy (CSP). *Scientific Culture*, *4*(2), 51-72.

Papert, S. (1993). *The children's machine: Rethinking school in the age of the computer*. BasicBooks, 10 East 53rd St., New York, NY 10022-5299.

Piaget, J. (1972). Development and learning. *Readings on the development of children*, 25-33.

Qian, M., & Clark, K. R. (2016). Game-based Learning and 21st century skills: A review of recent research. *Computers in Human Behavior*, *63*, 50-58.

Resnick, M., & Silverman, B. (2005). Some reflections on designing construction kits for kids. In *Proceedings of the 2005 conference on Interaction design and children* (pp. 117-122). ACM.

Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education*, *13*(2), 137-172.

SFIA Foundation. (2017). The Skills Framework for the Information Age – SFIA. Ανακτήθηκε στις 15 Μαρτίου, 2018 από: https://www.sfia-online.org/en/reference-guide.

Siegle, D. (2009). Developing Student Programming and Problem-Solving Skills with Visual Basic. *Gifted Child Today*, 32(4), 24-29. (ERIC Document Reproduction Service No. EJ860950).

Weintrop, D. (2015). Comparing Text-based, Blocks-based, and Hybrid Blocks/Text Programming Tools. In *Proceedings of the eleventh annual International Conference on International Computing Education Research* (pp. 283-284). ACM.

Weintrop, D., & Wilensky, U. (2015). To block or not to block, that is the question: students' perceptions of blocks-based programming. In *Proceedings of the 14th International Conference on Interaction Design and Children* (pp. 199-208). ACM.

Xinogalos, S., & Satratzemi, M. (2004). Introducing novices to programming: a review of teaching approaches and educational tools. In *Proceedings of the 2nd International Conference on Education and Information Systems, Technologies and Applications (EISTA 2004)* (pp. 60-65).

Zapušek, M., & Rugelj, J. (2013). Learning programming with serious games. EAI Endorsed Trans. Game Based Learn, 13(01), e6.

Ατματζίδου, Σ., Μαρκέλης, Η., & Δημητριάδης, Σ. (2008). Χρήση των LEGO Mindstorms στο Δημοτικό και Λύκειο: Το παιχνίδι ως έναυσμα μάθησης. Στο Β. Κόμης (επιμ.), *Πρακτικά 4ου Πανελλήνιου Συνεδρίου «Διδακτική της Πληροφορικής»*, Πάτρα: Εκδόσεις Νέων Τεχνολογιών.

Γρηγοριάδου, Μ., Γόγουλου, Α., & Γουλή, Ε. (2002). Εναλλακτικές Διδακτικές Προσεγγίσεις σε Εισαγωγικά Μαθήματα Προγραμματισμού. Στο Α. Δημητρακοπούλου (Επιμ.) *Πρακτικά Εισηγήσεων 3ου Πανελληνίου Συνεδρίου με Διεθνή Συμμετοχή «Οι Τεχνολογίες της Πληροφορίας και της Επικοινωνίας στην Εκπαίδευση»*, (Ρόδος, 26-29 Σεπτεμβρίου 2002), 239-248.

Ελευθεριώτη, Ε., Καρατράντου, Α., & Παναγιωτακόπουλος, Χρ. (2010). Χρησιμοποιώντας τα Lego Mindstorms NXT για τη διδασκαλία του Προγραμματισμού σε ένα διαθεματικό πλαίσιο: μία πιλοτική μελέτη. Στο Α. Τζιμογιάννης (Επιμ.), *Πρακτικά 7ου Πανελλήνιου Συνεδρίου με Διεθνή Συμμετοχή «Οι ΤΠΕ στην Εκπαίδευση»*, Πανεπιστήμιο Πελοποννήσου, Κόρινθος.

Εφόπουλος, Β. (2005). *Διαδικτυακό Περιβάλλον υποστηριζόμενο από Σύστημα Διαχείρισης Βάσεων Δεδομένων για την εισαγωγή στη διδασκαλία των αρχών του προγραμματισμού* (Διδακτορική διατριβή). Πανεπιστήμιο Μακεδονίας.

Εφόπουλος, Β., Δαγδιλέλης, Β., & Ευαγγελίδης, Γ. (2005). Η επιλογή της κατάλληλης εισαγωγικής γλώσσας προγραμματισμού για αρχαρίους. *Πρακτικά 3ου Πανελλήνιου Συνέδριου για την «Αξιοποίηση των τεχνολογιών της Πληροφορίας και της Επικοινωνίας στη Διδακτική Πράξη»*, Σύρος.

Καγκάνη, Κ., Δαγδιλέλης, Β., Σατρατζέμη, Μ., & Ευαγγελίδης, Γ. (2005). Μια Μελέτη Περίπτωσης της Διδασκαλίας του Προγραμματισμού στη Δευτεροβάθμια Εκπαίδευση με τα LEGO Mindstorms. Στο Α. Τζιμογιάννης (επιμ.), *Πρακτικά 3ου Πανελλήνιου Συνεδρίου «Διδακτική της Πληροφορικής»*, Πανεπιστήμιο Πελοποννήσου, Κόρινθος, 7-9 Οκτωβρίου 2005.

Κόμης, Β. (2005). *Εισαγωγή στη Διδακτική της Πληροφορικής*, Αθήνα: Κλειδάριθμος.

Κόμης, Β. (2004). Εισαγωγή στις εκπαιδευτικές εφαρμογές των Τεχνολογιών της Πληροφορίας και των επικοινωνιών, Αθήνα: Εκδόσεις Νέων Τεχνολογιών, Αθήνα

Ομάδα Εργασίας (2011). *Πρόγραμμα Σπουδών για τον Πληροφορικό Γραμματισμό στο Γυμνάσιο*. Αθήνα: Παιδαγωγικό Ινστιτούτο. https://goo.gl/eerBKk

Τζιμογιάννης Α., Πολίτης, Π., & Κόμης, Β. (2005). Μελέτη των αναπαραστάσεων τελειόφοιτων μαθητών Ενιαίου Λυκείου για την έννοια της μεταβλητής. Στο Α. Τζιμογιάννης (Επιμ.), *Πρακτικά 3ου Πανελλήνιου Συνεδρίου «Διδακτική της Πληροφορικής»,* 61-70, Κόρινθος.

Τζιμογιάννης, Α. (2015). Προς ένα Παιδαγωγικό Πλαίσιο Διδασκαλίας του Προγραμματισμού στη Δευτεροβάθμια Εκπαίδευση. Στο Α. Τζιμογιάννης (Επιμ.), *Πρακτικά Εργασιών 3ου Πανελλήνιου Συνεδρίου «Διδακτική της Πληροφορικής»* Πανεπιστήμιο Πελοποννήσου, Κόρινθος, 7-9 Οκτωβρίου 2005.

Φράγκου, Σ. & Παπανικολάου, Κ. (2010). Εκπαιδευτική αξιοποίηση συστημάτων ρομποτικής. Στο Γρηγοριάδου Μ. (επιμ) *Πρακτικά 5ου Πανελλήνιου Συνεδρίου Διδακτικής της Πληροφορικής, Τμήμα Πληροφορικής και Τηλεπικοινωνιών*, ΕΚΠΑ, Αθήνα, 9-11 Απριλίου 2010.

Χαρίσης, Χ. & Μικρόπουλος, Τ.Α. (2008). Ρομποτική, Οπτικός Προγραμματισμός και Βασικές Προγραμματιστικές Δομές. Στο Β. Κόμης (Επιμ.), *Πρακτικά 4ου Πανελλήνιου Συνεδρίου «Διδακτική της Πληροφορικής»*, Πάτρα: Εκδόσεις Νέων Τεχνολογιών.