

ΠΡΟΣ

- 1) Όλα τα μέλη ΔΕΠ του Τμήματος Επιστήμης Υπολογιστών
- 2) Τους εκπροσώπους των Μεταπτυχιακών φοιτητών του Τμήματος Επιστήμης Υπολογιστών
- 3) Την Επταμελή Εξεταστική Επιτροπή
- 4) Όλα τα μέλη της Πανεπιστημιακής Κοινότητας

Πρόσκληση σε Δημόσια Παρουσίαση της Διδακτορικής Διατριβής του

κ. Βαλσαμάκη Ιωάννη

Doctoral Dissertation Defense

Mr. Ioannis Valsmakis

Την Τρίτη, 13/04/2021 και ώρα 12:00, μέσω Τηλεδιάσκεψης, δημόσιας μετάδοσης (url<http://video.ucnet.uoc.gr/live/show/371>, YouTube channel του Τμήματος: https://www.youtube.com/channel/UC7uE3QiMTQjkrpByB_Gnt6Q/live στο Ηράκλειο, θα γίνει η δημόσια παρουσίαση και υποστήριξη της Διδακτορικής Διατριβής του υποψηφίου διδάκτορα του Τμήματος Επιστήμης Υπολογιστών κ. Βαλσαμάκη Ιωάννη με θέμα:

“ Ολοκληρωμένο Περιβάλλον Ανάπτυξης Για Οπτικό Προγραμματισμό: Εργαλεία Διαχείρισης, Συνεργασίας και Επεκτάσεων”

“Full-Scale Visual Programming IDE: Projects, Collaboration and Domain Plugins”

ΠΕΡΙΛΗΨΗ

Σήμερα, οι γλώσσες και τα συστήματα οπτικού προγραμματισμού είναι τα πιο δημοφιλή συστήματα για μη επαγγελματίες προγραμματιστές. Αρχικά εμφανίστηκαν για εκπαιδευτικούς σκοπούς, ως πειραματικά εργαλεία που ενθαρρύνουν τα παιδιά να προγραμματίσουν παιχνίδια μικρής κλίμακας. Τα εργαλεία οπτικού προγραμματισμού στις μέρες μας αντιμετωπίζονται όλο και περισσότερο ως προγράμματα που μπορούν να προσφέρουν πιο ισχυρές και ευέλικτες δυνατότητες διαμόρφωσης, προσαρμογής και επέκτασης σε χρήστες των συστημάτων λογισμικού, μέσω της ελέγξιμης δυνατότητας για προγραμματισμό που βασίζεται στην εξαγωγή λειτουργικότητας του εκάστοτε συστήματος. Μια

τέτοια προσέγγιση έχει ήδη εφαρμοστεί σε διάφορα συστήματα μεγάλης κλίμακας μέσω πλαισίων δέσμης ενεργειών, αλλά εξακολουθεί να απευθύνεται περισσότερο σε επαγγελματίες χρήστες και είναι πολύ απαιτητική για τον γενικό τελικό χρήστη.

Συνολικά, στην ταχέως αναδυόμενη εποχή μη προγραμματιστών (EUD) η υιοθέτηση γλωσσών οπτικού προγραμματισμού (VPL) φαίνεται πιο ελπιδοφόρα από ποτέ. Στην πραγματικότητα, η ευρεία διάδοση των τεχνολογιών για το διαδίκτυο των πραγμάτων έχει θέσει την ανάπτυξη εφαρμογών από μη προγραμματιστές ως το όχημα για να φιλοξενήσει τις αυξημένες απαιτήσεις για έξυπνους αυτοματισμούς. Συγκεκριμένα, ο τομέας του διαδικτύου των πραγμάτων (IoT) εξακολουθεί να αντιμετωπίζει χαμηλή εμπορική αποδοχή, κάτι που αποδίδεται στη χαμηλή δημοτικότητα των μονολιθικών και όλα σε ένα λύσεων. Είναι σαφές ότι υπάρχει μια τάση προς τις πιο ευέλικτες και ανοιχτές υποδομές που οι τελικοί χρήστες να μπορούν να τις προσαρμόσουν στις προσωπικές τους απαιτήσεις και ακόμη και να τις συνδυάσουν μέσα από νέους τρόπους με προσαρμοσμένους προγραμματίσιμους προσωπικούς αυτοματισμούς.

Ωστόσο, οι υπάρχουσες γλώσσες οπτικού προγραμματισμού υποστηρίζονται από πολύ πρωτόγονα και φτωχά σύνολα εργαλείων, χωρίς την έννοια ενός ολοκληρωμένου περιβάλλοντος ανάπτυξης (IDE) πλήρους κλίμακας με όλα τα εγγενώς απαιτούμενα σύνολα εργαλείων υψηλής ποιότητας. Υπό αυτήν την έννοια, οι λειτουργίες που λείπουν θα πρέπει να εστιάζουν ιδιαίτερα στην πραγματική βελτιστοποίηση της διαδικασίας προγραμματισμού για μη προγραμματιστές, πράγμα που σημαίνει ότι ο κατοπτρισμός της τυπικής λειτουργικότητας IDE του επαγγελματικού τομέα ανάπτυξης λογισμικού είναι ανεπαρκής και μάλλον ακατάλληλος.

Για τον σκοπό αυτό, ως μέρος αυτής της διατριβής θέτουμε μια μεγάλη πρόκληση: να ορίσουμε, να αναπτύξουμε και να επικυρώσουμε σε έναν πραγματικό και απαιτητικό τομέα τι θα πρέπει να προσφέρει ένα ολοκληρωμένο σύνολο εργαλείων για ανάπτυξη εφαρμογών από μη προγραμματιστές. Στα πλαίσια αυτά, η πρωταρχική πρόκληση ήταν η ανάπτυξη ενός πλήρους IDE για οπτικό προγραμματισμό, ικανό να φιλοξενήσει οποιοδήποτε συντάκτη οπτικού προγραμματισμού. Στην συνέχεια, η έρευνά μας, εμφάνισε και υποστήριξε έναν

αριθμό από βασικούς κλάδους στα πλαίσια εργαλείων προγραμματισμού για μη προγραμματιστές τα οποία και σχεδιάστηκαν, υλοποιήθηκαν και ελέγχθηκαν πλήρως στα πλαίσια του ολοκληρωμένου περιβάλλοντος ανάπτυξης για οπτικό προγραμματισμό: (i) υποβοηθούμενη διαχείριση έργων, (ii) εργαλείο συνεργασίας στα πλαίσια της επεξεργασίας και του εντοπισμού σφαλμάτων και (iii) ανοιχτοί διαδραστικοί προστιθέμενοι τομείς εφαρμογών.

Συγκεκριμένα, ο ρόλος των προστιθέμενων τομέων είναι πολύ κρίσιμος, πολύ περισσότερο από απλά πακέτα επέκτασης, με σκοπό τον εμπλουτισμό της διαδραστικής λειτουργικότητας του IDE με επιπλέον δυνατότητες ανάπτυξης που ταιριάζουν βέλτιστα σε έναν συγκεκριμένο τομέα εφαρμογών. Αυτή η έννοια είναι νέα για τα εργαλεία για μη προγραμματιστές, χωρίς αντίστοιχη στα παραδοσιακά ολοκληρωμένα προγραμματιστικά περιβάλλοντα για επαγγελματίες προγραμματιστές, και στοχεύει να αντιμετωπίσει την εγγενή πολυπλοκότητα των τομέων εφαρμογών για τα εργαλεία για μη προγραμματιστές λόγω των εξειδικευμένων μοντέλων προγραμματισμού και των βιβλιοθηκών που είναι πολύ δύσκολο να διαχειριστούν χωρίς επιπλέον εργαλειοθήκες πάνω από το IDE. Για να δοκιμάσουμε και να επικυρώσουμε την πρότασή μας, έχουμε αναπτύξει, πάνω από το ολοκληρωμένο περιβάλλον οπτικού προγραμματισμού, ένα πλήρες πρόσθετο τομέα εφαρμογής για το διαδίκτυο των πραγμάτων, που περιλαμβάνει ένα πολύ πλούσιο διαδραστικό σύνολο εργαλείων, για υποστήριξη των μη προγραμματιστών έτσι ώστε να τους ενθαρρύνει να προγραμματίσουν τους προσωπικούς τους έξυπνους αυτοματισμούς.

Επιβλέπων: Καθηγητής, Αντώνης Σαββίδης

ABSTRACT

Today, visual programming languages (VPLs) are the most popular programming system for non-professional developers. Originally, they have been introduced for teaching purposes, as experimental tools encouraging children to program small-

scale games. Nowadays, they are increasingly treated as instruments that can give more powerful and flexible configuration, customization and extension features to the end-users of software systems, through controllable programmability relying on some exposed underlying system functionality. Such an approach has already been applied within various large-scale systems via scripting frameworks, but is still targeted to more professional users and is very demanding for the general end-user.

Overall, in the rapidly emerging era of end-user development (EUD) the adoption of VPLs seems more promising and hotter than ever before. In fact, the broad proliferation of Internet of Things (IoT) technologies has set end-user development as the vehicle to accommodate the increased personalization demands for smart automations. In particular, the IoT domain still faces a low commercial acceptance, something attributed to the low popularity of monolithic and all-in-one solutions. It is clear that there is trend towards more flexible and open infrastructures that end-users may directly tailor to their individual requirements, and even functionally combine into new ways with custom-made programmable personal automations.

However, the existing VPLs are supported with very primitive and poor tool chains, missing the notion of a full-scale integrated development environment (IDE) with all the inherently required high-quality production toolset. In this sense, the missing features should be explicitly focused on genuinely optimizing the end-user programming process, meaning the mirroring of typical IDE functionality of the professional software development domain is insufficient and rather inappropriate.

To this end, as part of this thesis we set one grand challenge: *define, develop and validate in a demanding real domain what an integrated toolset for end-user development should offer*. To this end, the primary technical challenge has been the development of a full-scale IDE for VPLs, capable to accommodate and host virtually any VPL editor. Then, our research has revealed and supported a number of primary disciplines in the context of EUD that we have fully designed, implemented and tested in the context of our IDE: (i) *assisted project management*, (ii) *collaborative editing and debugging*, and (iii) *open interactive domain plugins*.

In particular, the role of the domain plugins is very crucial, far more than mere extension packages, with the purpose of enriching the interactive IDE functionality with extra development features optimally suiting a target domain. This notion is novel to EUD, with no counterpart in traditional IDEs, and aims to address the inherent complexity of domains for EUD due to the custom programming models and libraries that are very hard to manage without extra toolboxes on top of the IDE. To test and validate our proposition we have developed, on top of our IDE, a complete full-scale IoT plugin, including a very rich interactive toolset, for EUD support of personal smart automations.

Supervisor: Professor, Antonis Savidis