

# Psychosocial Correlates of Software Designers' Professional Aptitude

Walery Susłowski, Jacek Kowalczyk, Michał Statkiewicz  
Koszalin University of Technology  
Koszalin, Poland

Marta Boińska, Janina Nowak  
The University of Gdansk  
Gdansk, Poland

**Abstract**—This paper presents quantitative results of the first phase of empirical research carried out within the framework of the interdisciplinary project InfoPsycho that was initiated in 2013 at the Koszalin University of Technology and the University of Gdansk. The aim of the study was to identify the personality traits that characterize successful applicants for university studies in the field of software development. Synthetic indicators of quality and performance of their design tasks and exercises were selected as the criteria for candidates' professional skills. To measure personality traits, the NEO-FFI questionnaire was used, based on the five-factor model by Costa and McCrae. Preliminary results show that expected young designers (N=140) score high on neuroticism and introversion as compared with those designers whose design documentation is of poor quality. They also show a high degree of conscientiousness, which can be seen when their performance of exercises and programming tasks is being evaluated.

**Keywords**—Software designer; Software quality; Professional skills; Early diagnosis; Psychological tests; NEO-FFI

## I. INTRODUCTION

One of the major challenges facing the education system in Poland is to adapt the university educational offer to the market needs. The number of software developers is constantly increasing while the demand is not diminishing but it is rather growing consistently and steadily. Hence, an important task seems to be to examine the psychological correlates of career aptitude in this profession. These correlates should be good indicators of future optimum efficiency of the designer and of their software products commercial success.

These questions have been present in literature for a relatively short period. Despite increased research into psychological factors in the field of computer science in many countries, we cannot detect this wider trend in Polish literature. This seems all the more surprising given that IT experts in the area of Central and Eastern Europe are considered to be one of the most successful software engineers in the world.

The result of developers' teamwork is software, which is usually ordered from them by a customer who possesses relatively little knowledge of programming languages or the latest solutions and trends that emerge in the field. An average customer has little experience in ordering software solutions. The end-users only have limited knowledge of what they would like to order without knowing the vast possibilities of how the ultimate product can be specifically designed to suit their needs. Thus, from a practical point of view, it is important that the software designer (the leading specialist in the

development team) possesses high-level communication skills and specific personality traits that support the product creation process during which the customer is a powerful decision-maker. One could say that the software designer should possess both remarkable interpersonal and technical skills to perform their job effectively. This is an assumption based on an important observation that led us to undertake joint efforts with researchers in the area of social sciences and information technology with the aim of designing an interdisciplinary assessment programmers' advanced academic and workplace performance using various psychosocial variables. The research should be interdisciplinary in nature for two reasons: a professional evaluation of developers' suitability can only be performed by highly qualified specialists in the field of computer science, while an accurate assessment of psychosocial variables can be carried out only by psychologists who are willing to draw up and analyze opinion questionnaires.

Modern methods of software development transform a set of user requirements related to data processing into a set of instructions (a computer program within the desired framework). The common feature of these methods is that they are based on a creative activity using a design conception, not on a ready template. Hence, software quality is significantly influenced by designer's intuition and experience. By using the expressions used in literature on project typology already published [1], we can conclude that designer's personality traits will be absolutely critical in the analytical and design stage of the project (according to the criterion of the life cycle phases) as well as the primary and development stage of the project (according to the criterion of changes).

Developing IT projects is a very demanding job because the designer ought to be an expert in information technology, know which efficient and reliable methodology to adopt, and possess a good understanding of the specific problem domain. In practice, hard systems methodologies are used where the developer first comes up with a technical design and the whole responsibility is shifted onto the designers or programmers who develop and launch a product in close cooperation with the customer. They gain the necessary knowledge about the problem area and search for a comprehensive solution that is technically achievable and most user-friendly. The most dramatic development of the software systems takes place in small IT groups which have a fairly limited budget and an insufficient number of expert advisors. Unfortunately, these bad conditions often force the programmer to act not only as the architect of the system but also as a specialist in other areas. Similar problems occur in major projects that are poorly

defined and coordinated, where the Open Source projects are a leading example.

For commercial software to be successful, it should be a high quality product. That is not a question of understanding the quality in terms of the degree of perfection as Plato defined it. Good software must be reliable, stable, efficient, safe, ergonomic, and portable. However, software is of outstanding quality when it is in accordance with the requirements imposed by the project framework and it meets all the user's needs and expectations. Thus, the buyer's satisfaction with the software, which takes place at the stage of deployment of the finished product, is in essence an evaluation of the product's quality. This forces designers to confront their own personal standards and ideas with the attitudes, needs and expectations of a broadly understood consumer.

An analysis of the specificity of the software as a market product leads to the conclusion that in order to help students excel in one specialized area of software design, the academic training programme should involve teaching the know-how about dealing with service clients and product end-users. It is commonly believed that the profession of the IT specialist involves only the technical background knowledge and necessary problem-solving skills. Indeed, programming skills and good knowledge of modern information technology, and hardware platforms are a firm foundation for this vocational career. But this is not enough; the designer needs a good "interface" as well as communication and cooperation skills to be able to work in a team and with a wide range of stakeholders. He should be willing to learn about innovations, to widen his perspective, and adapt his point of view to the ever-changing market trends. In other words, apart from being equipped with excellent technical knowledge, the software programmer should master humanistic skills (social, psychological, emotional). Unfortunately, this aspect of the vocational training of software designers is missing from the curricula of technical universities. As to some of the humanistic skills, it can be assumed that they cannot be effectively acquired by all students of computer science without guidance. What is now a common practice is that university teachers can only try to choose students according to their psychosocial aptitudes, and assign them specializations that seem to most suitable for their future career.

## II. REVIEW OF EXISTING LITERATURE

Research on psychosocial variables among software programmers makes use of a variety of theoretical models and psychometrics with different effects. The sections to follow provide a brief description of psychometric instruments and models, deliver the theoretical framework of the study, and then summarize the current results of research on the personality of programmers.

### A. Models of personality and their operationalization in research on software developers' personality traits

Recently, there has been a great interest in testing and evaluating psychosocial variables, including personality, among the group of programmers [2-6]. The techniques used in these studies, which are very diverse, are the effect of an operationalization of a particular theoretical model that

investigates personality [7-8]. There is a need to deepen and replicate research for the population of programmers.

The most common method of measuring personality traits in studies on programmers has been so far the Myers-Briggs Type Indicator (MBTI) [9-11]. It is based on Carl Gustav Jung's theory of psychological types of individuals. The MBTI questionnaire covers four bipolar factors or dimensions: extraversion-introversion (EI), sensing-intuition (SN), thinking-feeling (TF), and judging-perception (JP). An equally widely-used personality assessment was the Keirsey Temperament Sorter (KTS) questionnaire [12]. However, more recent studies show a shift to a five-factor model of personality known in short as Big-Five [13-14]. Psychologists know that, for testing purposes, the questionnaire to measure personality traits should be selected based on a number of factors, which have been defined by the methodologists of psychological research. The most important parameters of a good questionnaire are reliability and validity. The MBTI questionnaire, based on the concept of Jung's psychodynamic approach to personality, has been criticized despite its popularity because of the above-mentioned main parameters [15]. The results obtained by this method provided an obscure picture of programmers' personality. Hence, it is recommended to explore a different theoretical model and, consequently, other research techniques [16-18]. Due to the above-mentioned psychometric limitations of the MBTI questionnaire, in presented study the five-factor model of personality (PMO) designed by Paul Costa and Robert McCrae is employed. This model is used more frequently; what is more, it has repeatedly been tested in academic research and clinical trials [19]. It differentiates individuals in a significant way, which is widely regarded [20-22]. It represents a hierarchically well-organized personality inventory of five characteristics: extraversion, agreeableness, conscientiousness, openness and neuroticism. Here is a brief description of these traits:

1) *Extraversion (E) indicates the degree to which a person is sociable, assertive and active in a conversation. An extrovert feels good in social relationships and derives pleasure from these.*

2) *Agreeableness (U) refers to individual characteristics such as kindness, trust and warmth. A person with a low level of agreeableness is described as selfish and full of doubts towards society.*

3) *Conscientiousness (S) deals with one's orientation to achievements. People who receive high scores are hardworking, reliable and organized enough to perform their tasks on time. On the other hand, low scores for conscientiousness are characteristic of impulsive, disordered, and irresponsible people.*

4) *Openness (O) describes the readiness of individuals to a wide range of intellectual and cultural activities. A person with high openness, one that has broad horizons, is ready to take risks to stimulate. At the opposite extreme, there is a person with low openness to experience, showing little sensitivity to aesthetic or cultural stimulation.*

5) *Neuroticism (N) alternatively described as an emotional stability [23], is often correlated with a sense of*

*self-efficacy [24]. People with low score are confident; feel safe, relatively stable in mood, calm. High neuroticism is an indicator of frequently changing moods, nervousness, anxiety and uncertainty [25].*

The rationale for popular PMO methods is lexical hypothesis, which is one of the key assumptions of the model. It points out that those five attributes (factors) described are contained in a natural language, one which we use in our country. These factors can be considered as the elements of our own "universal consciousness" as they concern the most adaptive characteristics and individual differences between people [26]. As any theoretical model, the PMO has its limitations and it has received some critical opinions, mainly related to the amount of factors that describe the personality. Nevertheless, the PMO is well studied in the academic and practical field, what makes this model of personality a standard choice of operationalization to numerous studies [27]. PMO is intended for testing many different tools, such as NEO-PI-R, NEO-FFI, and IPIP. NEO-FFI is an instrument which is used in this study due to its reliability and validity and due to its common usage to measure the personality in academic field [28-29]. Careful analysis of the literature allowed to make a decision to input PMO as a theoretical framework, which is operationalized in the form of the NEO-FFI instrument. The section to follow describes a review of the studies published, ones which focused on relations between the personality and the programmer's achievements.

#### *B. Research on computer programmers' personality and performance*

There are many studies in literature that test relations between the programmer's psychosocial qualities and the indicators of its operation. The main areas of studies are related to the issues of programming in pairs (PIP), as well as the efficiency of development teams (EMPA). Unfortunately, the results of these studies do not provide clear answers about the relationship between programmer's personality and the quality of his programming outcomes [15].

In this section three studies are presented where researchers were testing the relations of the personality of programmers and the efficiency of programming outcome in PIP. The results obtained do not always confirm the hypotheses. Only one study has fully demonstrated that in the process of programming in pairs a proper fit of personalities has resulted in the improvement of effectiveness [17]. In this publication, Choi states that if two people, without any previously acquired programming experience, are similar in terms of major personality traits or the MBTI complementary model (ST-SF, NT-NF, ST-NT, and SF-NF), their level of performance (in terms of quality of the software created) will be much higher than for other combinations. Another study [31] suggests that different levels of conscientiousness (understood as a personality trait) do not affect the academic success of paired programmers-students in the course of joint programming. Research [12], which was used in KTS, suggests that pairs consisting of a heterogeneous personality better fall within the parameters of the programming rather than a pair of the same type of personality. Most studies have provided inconsistent results, or ones that do not confirm the existence of any significant differences in terms of personality in PIP.

Equally often tested is how programmer team is managed and what influences the efficiency of team members. The most important study in this area is one where an attempt was made to determine the impact of personality on the results of team programming. Research has shown that teams can work in a satisfactory manner, despite significant differences in the members' personalities and ethnic and religious backgrounds [33]. On the other hand, there are the results showing that in fact there is a significant correlation between personality factors and the satisfaction of teamwork [32]. In contrast, Chao and Atli [10] studied the personalities of 60 respondents, and the data obtained by them did not allow to confirm the thesis that there is a difference in the quality of code between different matching types of personality in pair programming tasks. During the literature review, it is difficult to find accurate data on the personality of programmers in the context of individual work. Few studies reports show inconsistent results. Capretz L. F. [7, 10] shows that according to the classification of the Jung's MTBI model, among programmers there are 57% introverts vs. 43%, extroverts 81% minded vs. 19% sentient, 58% judgmental vs. 42% followers. The predominant set of traits is an introvert personality ISTJ (introversion, sensing, thinking, judgment): 24% for the entire sample of the respondent programmers, where studies on the general population of computer ISTJ reach only 11.6%. People of this type are "systematic, robust, practical, realistic approaching to reality, keeping promises, respecting the duties, valuing facts, well-organized, selfless" [34].

The question is whether such a set of traits is beneficial for the quality of the software developed. Cunha and Greathead [11] attempted to find an answer to this question by examining students in terms of the severity of the characteristics of the MBTI model. It turned out that people with higher results for the dimension of intuition performed significantly better in a task requiring a review of 282 lines of the Java code. In turn, Acun with colleagues indicates that extraversion is significantly associated with a better software quality in software development in an agile methodology [35].

### III. REPORT ON THE PROGRESS OF THE INFOPSYCHO PROJECT

#### *A. Concept and hypothesis*

The aim of the present study was to examine whether and how psychological predispositions (personality traits) of young software developers, recorded in the early stages of their professional education, are associated with an ability to develop good, user-friendly client software, one that fulfills the customer's requirements. Since the literature does not allow us to pose specific research hypotheses, this study is of an exploratory (pilot) nature. We pose the following research question: "What personality traits characterize young designers who are successful in the field of software at university level?" We have set two general hypotheses:

- 1) There is a relationship between personality traits and the quality of software created by a designer.*
- 2) It is possible to indicate those personality traits that are important for the quality of design of software before a young designer commences their professional career.*

To verify the abovementioned hypotheses, we tested two different age groups of students-informaticians taking into account different performance indicators of students' work on the software exercise mode and the design mode. To ensure the anonymity of the test, we used codes that allowed us to combine the results of individual indicators without identifying any individual participants registered.

We posed one additional research question, which was not followed by a hypothesis, because the scientific literature does not provide a sufficient suggestion for a probable answer: Do high-performing and low-performing IT students differ from non-technical faculty students in personality traits? We investigated personality traits of Pedagogy students and compared their results with IT students achieving the lowest and the highest outcomes in project quality scale in the second group.

### B. Groups of participants

The group examined in the project InfoPsycho included 128 men and 12 women who were undergraduate and postgraduate students of Computer Science in the Department of Electronics and Computer Science at the Koszalin University of Technology. The structure of pilot studies carried out in the period from November 2013 to February 2014 assumed a division of the respondents into two separate groups. The first subgroup included 65 people (one person was later excluded because of missing some data), where men constituted 89.4%. They were second-year students actively involved in computer programming and information technology, but still not involved in design work. The average age for this subgroup was 22 years old. The second subgroup included 73 people (one person was later excluded because of missing some data), where 93.2% were men. This subgroup included students of the final year of undergraduate and postgraduate studies. They were all project contractors and, during the experiment, they performed the role of software designers, analysts, and software project managers. The average age in this subgroup was 24 years old.

### C. Research techniques and tools

NEO-FFI Costa and McCrae questionnaire in the Polish adaptation by Zawadzki, Strelau, Szczepaniak and Sliwinski was used as a psychological tool [32], measuring five dimensions of personality in the Costa and McCrae concept: neuroticism, extroversion, openness to experience, agreeableness and conscientiousness. The questionnaire included 60 positions – 12 for each of 5 subscales. The test optional answers were presented on the five-point Likert's scale ranging from "strongly disagree" to "strongly agree".

The abovementioned personality dimensions were measured in the form of a self-report inventory. The results represent a full description of the personality of participants and the anticipation of their adaptability to the professional environment [32]. Time given to fill in the questionnaire was approximately 15 minutes.

Indicators that characterized the quality of design<sup>1</sup> done by the students were brought to a three-level rating scale; 1-2-3 is ranging from the worst to the best. When the exercise programming, the software system design, and the quality of communication in the project's team was judged, the expert approach was used; judgment was aided by checklist concerning all necessary elements of the students' work.

The general description of the specific nature of development work and the idea of quality of software are presented in the introduction to this article. Quality of the students' design was assessed with respect to a set of guidelines:

- compliance with formal requirements (adequate description, sequence of activities, presence of diagrams) – a critical guideline;
- syntactic correctness – a critical guideline;
- conceptual correctness (consistency of design and functionality of the designed system) – a critical guideline;
- efficiency and originality of created solutions;
- readability of the design documentation (describing and diagramming style, presence of glossary);
- history of the project (presence of schedule the number of patches).

It is necessary that the critical guidelines are met. The degree to which students follow the guidelines determines the final result of the assessment. However, when assessing the programming exercises, we have taken into account the following aspects of the students' work:

- correctness of the realized task (solution to the problem in the form of a compilable source code) – a critical guideline;
- good understanding and correct analysis of the code – a critical guideline;
- use of good programming practices (object-oriented paradigms, readability, error handling);
- efficiency of technical solution.

Students' activity assessment was based on active participation and involvement in meetings with supervisors and student coordinators when students' solutions and decisions were consulted. The parameter of the periodic activity of each project group was calculated in proportion to the number of meetings and of the closed tasks, and then a numerical value of the range of 1-2-3 was assigned.

### D. Results

The results are presented in two main stages. First, we show correlation coefficients between two groups: the grade,

---

<sup>1</sup> "Level of effectiveness of the design function in determining a product's operational requirements (and their incorporation into design requirements) that can be converted into a finished product in a production process", definition from <http://www.businessdictionary.com>.

activity during the lessons, and personality traits in the first subgroup of IT students and quality of design, communication quality, and personality traits in the second subgroup of IT students. Second, we compare three set of data: high-performing IT students from the second subgroup, low-performing IT students from the second subgroup, and Pedagogy students. In both stages, the alpha level of 0.05 was applied. Statistical analyses were conducted with IBM SPSS Statistics 21.

Table I presents the means, standard deviations, and correlation coefficients of the grade, activity during lessons, and the personality traits of the first subgroup of IT students. We use two different correlation coefficients to explore relationships between different types of scales: Kendall's tau-b for personality traits (interval scale), grade (ordinal scale) and activity during the lessons (ordinal scale); Spearman's Rho for grade (ordinal scale) and activity during lessons (ordinal scale).

TABLE I. MEANS, STANDARD DEVIATIONS, AND CORRELATION COEFFICIENTS BETWEEN THE GRADE, ACTIVITY DURING LESSONS, AND PERSONALITY, N=65

	M	SD	1	2	3	4	5	6	7
1. Grade			1.00	0.70**	-0.05	0.38**	-0.16	0.99	0.01
2. Activity during lessons				1.00	-0.07	0.21*	0.01	0.00	0.10
3. Extraversion	39.51	7.41			1.00	0.16*	-0.29**	0.23**	0.09
4. Conscientiousness	42.28	7.95				1.00	-0.18*	0.13	0.01
5. Neuroticism	32.05	9.18					1.00	-0.20*	0.03
6. Openness to experience	38.57	6.87						1.00	-0.14
7. Agreeableness	38.68	4.52							1.00

\* p < 0.05 \*\* p < 0.01

The grade correlates positively with conscientiousness (r = 0.38, p < 0.01). Also, activity during lessons correlates positively with conscientiousness (r = 0.21, p < 0.05). The grade correlates positively with the activity during lessons (r = 0.70, p < 0.01), which is consistent with the evaluation criterion: the common variance is the result of the number of correct answers given by each student. This is included in both variables.

The quality of design correlates positively with neuroticism (r = 0.21, p < 0.05) and negatively with extraversion (r = -0.25, p < 0.01). The communication quality correlates positively with agreeableness (r = 0.27, p < 0.01) and negatively with extraversion (r = -0.19, p < 0.05).

The remaining correlation coefficients failed to gain statistical significance. The quality of design correlates positively with the communication quality (r = 0.60, p < 0.01).

Table II presents the means, standard deviations, and correlation coefficients of the quality of design, the communication quality, and the personality traits of the second group of IT students. Similarly to the first group, we use two different correlation coefficients to explore the relationships between different types of scales: Kendall's tau-b for personality traits (interval scale), the quality of design (ordinal scale), and communication quality (ordinal scale); Spearman's Rho for the quality of design (ordinal scale) and the communication quality (ordinal scale).

In the next step, we compare high-performing IT students from the second subgroup, low-performing IT students from the second subgroup, and Pedagogy students. The groups do not differ significantly in age and gender. Means and standard deviations in personality traits in all three groups are presented in Table III.

TABLE II. MEANS, STANDARD DEVIATIONS, AND CORRELATION COEFFICIENTS BETWEEN QUALITY OF DESIGN, COMMUNICATION QUALITY, AND PERSONALITY, N=73

	M	SD	1	2	3	4	5	6	7
1. Quality of design			1.00	0.60**	-0.25**	0.08	0.21*	0.04	0.04
2. Communication quality				1.00	-0.19*	0.12	0.11	0.03	0.27**
3. Extraversion	39.70	6.78			1.00	0.21**	-0.29**	0.05	0.12
4. Conscientiousness	44.21	8.05				1.00	-0.25**	0.14*	0.22**
5. Neuroticism	29.16	7.95					1.00	0.50	-0.13
6. Openness to experience	37.97	6.03						1.00	-0.00
7. Agreeableness	45.14	6.89							1.00

\* p < 0.05 \*\* p < 0.01

TABLE III. MEANS AND STANDARD DEVIATIONS IN PERSONALITY TRAITS IN PEDAGOGY STUDENTS, HIGH-PERFORMING IT STUDENTS, AND LOW-PERFORMING IT STUDENTS

	Pedagogy students, N = 30		High-performing IT students, N=31		Low-performing IT students, N = 18	
	M	SD	M	SD	M	SD
Extraversion	42.50	7.09	38.13	6.94	42.78	5.91
Neuroticism	31.17	8.52	30.74	8.52	25.22	6.22
Conscientiousness	45.00	6.01	45.42	6.82	42.53	8.40
Agreeableness	40.83	5.72	42.00	6.03	42.33	5.51
Openness to Experience	37.63	6.22	37.96	6.40	37.33	5.85

A one-way comparison between ANOVA subjects was drawn to compare the personality traits of three groups of students. There is a significant difference between the groups in levels of extraversion [ $F(2, 76) = 4.11, p = 0.020$ ],  $\eta^2 = 0.43$  and neuroticism [ $F(2, 76) = 3.54, p = 0.034$ ],  $\eta^2 = 0.39$ . Remaining comparisons are insignificant: conscientiousness [ $F(2, 76) = 1.03, p = 0.361$ ], agreeableness [ $F(2, 76) = 0.479, p = 0.621$ ], openness to experience [ $F(2, 76) = 0.062, p = 0.940$ ]. Post hoc comparisons using the Tukey HSD test for neuroticism indicate that the mean score of the low-performing IT students is of minor significance as compared to the high performing IT students ( $I-J = -5.7, p = 0.054$ ) and is significantly different than that of Pedagogy students ( $I-J = -5.94, p = 0.044$ ). However, the high-performing IT students do not significantly differ from Pedagogy students ( $I-J = -0.23, p = 0.993$ ). Post hoc comparisons using the Tukey HSD test for extraversion indicate that the mean score of the high-performing IT students is of a borderline significance with the low-performing IT students ( $I-J = -4.65, p = 0.060$ ) and is significantly different than that of Pedagogy students ( $I-J = -4.37, p = 0.037$ ). However, the low-performing IT students do not significantly differ from the Pedagogy students ( $I-J = 0.28, p = 0.990$ ).

All in all, these results suggest that the high-performing IT students have a similar level of neuroticism and a lower level of extraversion than the Pedagogy students, whereas the low-performing IT students have a lower level of neuroticism and a similar level of extraversion as Pedagogy students.

#### E. Discussion of research findings

The results of the study confirm the first hypothesis. Achieving good grades in informatics correlates with specific dimensions of personality. A positive correlation was observed between the indicators of the quality of design and the level of neuroticism and introversion in terms of Costa and McCrae. Students with the highest degree of efficiency in the area of software design are more neurotic (5 sten) than students with a lower degree (3 sten). Differences become significant in intragroup comparisons; however, all the results fit in the given age norms. Intragroup differences in extraversion were also observed: students getting better results in quality assessment and communication were less extraverted (5 sten) than students getting worse results (6 sten). Students with the highest degree of efficiency in the area of software design are less emotionally stable than less predictive students in this field. Differences in extraversion were also observed: students getting better results in the assessment of the quality of design and communication were less extravert (5 sten) than students getting worse results (6 sten).

Many previous studies have shown that the level of neuroticism is correlated with the experience of negative emotions and emotional instability [33-35]. But higher levels of neuroticism also show that the individual is able to think in an unconventional way and comes up with very innovative ideas [36]. This seems especially desirable for future software designers whose significant advantage is the ability to "break" the generally applicable schemes (where appropriate; however, not ignoring reliable methodologies and solutions), even if such thinking seems at first incomprehensible or even illogical [37]. Much attention has been devoted to neuroticism by Karen

Horney [38] who described a model of neurotic competition. The components of such neurotic competition are as follows: always comparing oneself with others, the desire to be unique, and an element of hostility. The best work results and most original ideas very often are produced when one works in solitude and avoids the distraction of other people. People who prefer to work individually than work in a team are often deeply convinced that only few will succeed in a given field, therefore they steer clear of any collective work which can be even harmful. Neurotic independence, mentioned above, should be considered as a trait of a future employee in their workplace. On the one hand, a competitive individual will get the project implemented quickly and efficiently, but on the other hand, he can, for the same reason, have an opinion of a difficult employee. Salgado meta-analysis [39], where counterproductive behavior was the focal point, showed that emotional stability, which is the opposite of neuroticism, is associated with a lack of rotation in the workplace.

The combination of the results mentioned above, related to neuroticism and introversion among future successful programmers, is consistent with similar studies carried out in this convention [40]. This clearly suggests that IT specialists are more timid in their dealings with other people. Human relations for people with a high degree of introversion are strenuous, and therefore they spend more time working on their own. They prefer solitude and seem to have less need for the entertainment and fun that is often associated with working with peer colleagues. Each of their statement is scrupulously analyzed before it is said out loud [41]. It should be noted that our target group in this study were students who are not sufficiently aware of the significance of interpersonal relationships. The curriculum of computer science studies does not include any activities aimed at developing soft (social) skills. The academic work of these students is evaluated by other technical specialists, which results in students functioning in a rather hermetic environment of the university campus.

One of the qualities that is conducive to developing essential communication skills is a high level of agreeableness. This is in line with the previous research results. The results achieved suggest that individuals with a high level of agreeableness are able to work with others and can compromise [42]. They are also empathetic, friendly, and sympathetic [43]. Agreeableness components include dimensions such as straightforwardness, altruism, submissiveness, humility, a tendency of self-pity [44]. People with a higher level of agreeableness have a natural tendency to neutralize conflicts and emphasize the benefits that stem from functioning in a group [47]. It is also worth mentioning that people with high scores in this area are able to control their anger, which can also have a temperamental basis and foster proper communication [45].

A correlation between conscientiousness and the assessment of activities is consistent with studies which indicate that academic achievements are positively related to this exact personality dimension. Furthermore, a significant amount of research demonstrates that high levels of conscientiousness allow one to predict the functioning of the individual in the workplace [46-47]. Such a person is perceived as responsible, persistent in pursuing goals, thoughtful in

planning and undertaking new tasks. Low conscientiousness is associated with a tendency to do activities which may be described as procrastinative [48]. Conscientious employees are more credible, more motivated, they also have lower rates of absence and are less prone to harmful behavior at work, such as theft and aggressive behavior towards other workers [49].

Another issue, presented in the study, was a comparison of the students of Pedagogy with future software designers. The results showed that promising designers are characterized by a similar degree of neuroticism as the students of Pedagogy, while future IT specialists with a low degree achieve significantly poorer results in this area. This result leads to conclusion that a lower level of neuroticism is detrimental to the quality of their design. More neurotic individuals are more likely to interpret their mood adequately, analyze themselves, and look critically at reality, which may be associated with a lower self-esteem and a sense of helplessness. They are aware that improving their mood must have a real basis and is not the result of wishful thinking or the actual needs to be happy. Students of Pedagogy, who wish to follow the career of teachers and educators, are trained in the tasks of teaching others the values and traditions, as well as attitudes which tend to change over time. They learn how to teach openness and tolerance of cultural and individual differences which are of utmost importance in today's world. This valuable training leaves them less opinionated and more accepting of others.

In addition, a more realistic and critical view of the reality can be as well a good side of those individuals whose work is based on tedious and time-consuming programming, which depends on the final result. Such job often involves revision, reorganization of work, or verification of the details that are unfamiliar (mysterious) to others; it can lead to a short-term or long-term reduction in self-esteem; the individual will be determined to do his job properly and accurately.

What is more, promising software designers are characterized by a lower degree of extraversion than those with lower rates, or than students of Pedagogy. The pedagogue's future job, independently of the specialization completed, is associated with numerous interpersonal relationships. Pedagogical studies are focused on the development of social competence. Therefore, a high level of extraversion is the key to their career success. However, it turns out that a high level of extraversion, which for pedagogues can be an important indicator of their suitability for the profession, for software designers is much less of a desirable trait and can even impair the quality of their design. As mentioned earlier, promising software designers prefer to operate on their own, away from various distractors. When one functions alone, work is more predictable and under control. If software needs to be fixed, then only the designer can fix it quickly and safely. His work results are verified on regular basis. Such an operating mode seems to be particularly desirable in two cases: when working on individual IT projects, as well as when doing work in the remote mode.

To sum up, the analysis above suggests that promising software designers are characterized by specific personality traits. They are more introvert than Pedagogy students and future IT specialists who are not considered to be the

promising ones. They also score lower on neuroticism than future pedagogues, but higher than students who are not likely to specialize in the area of software design.

It should be emphasized that our study was of a pilot nature. Relatively small groups were a limitation to the study and, because of the university's educational process, these groups were broken down according to different evaluation criteria. In the next stages, our team will aim to set the evaluation criteria for the performance of IT specialists as accurately as possible. In addition to this, we will research more into the personality characteristics found in the dissertations of Costa and McCrae's on neuroticism, conscientiousness, and extraversion. This will allow us to build a more complete and detailed model of efficient programmer's personality. Apart from this, it will be necessary to expand the research to include also those designers who work with software professionally. This will provide evidence and be a starting point for discussion on how personality characteristics indicated by us, can be also diagnostic in the work environment.

#### IV. CONCLUSIONS

The results obtained in the pilot study indicate directly the existence of psychosocial correlates of professional predispositions of software developers. Following a preliminary research, it is already obvious how further research should be designed to lay a methodological basis for the practical use of psychological tests. As explained in previous passages, psychological tests support the learning process at the faculty of Computer Science. One would expect the implementation of the dynamic specializations' information management at faculties based on the commitment to the professional aptitude, to have already been discovered by the students of the second year, to their preferred roles in industrial design teams. In fact, this requires appropriate planning expertise based solely on the mating segments of the software market or platform implementation. The ability of a pro-active psychological profiling of vocational students (dynamic specializations), based on the forecast of the demand for IT professionals, can become answer to the need of adapting the educational offer of higher education to the market needs. A practical implication of the study would improve the communication skills of students-specialists. One aspect to consider is a possibility of a more efficient use of teaching hours planned in the humanities module.

#### REFERENCES

- [1] S. Wrycza, *Informatyka ekonomiczna*. PWE, Warszawa 2010, s. 345.
- [2] M. V. Kosti, R. Feldt, L. Angelis, Personality, emotional intelligence and work preferences in software engineering: An empirical study. *Information and Software Technology*, 56 (8): 973–990, 2014.
- [3] R. Feldt, R. Torkar, L. Angelis, M. Samuelsson, Towards individualized software engineering: empirical studies should collect psychometrics. In: *Proceedings of the 2008 International Workshop on Cooperative and Human Aspects of Software Engineering*, pp. 49–52, ACM, May 2008.
- [4] J. E. Hannay, E. Arisholm, H. Engvik, D. I. Sjoberg, Effects of personality on pair programming. *IEEE Trans. Softw. Eng.*, 36 (1): 61–80, 2010.
- [5] R. Feldt, L. Angelis, R. Torkar, M. Samuelsson, Links between the personalities, views and attitudes of software engineers. *Inf. Softw. Technol.*, 52 (6): 611–624, 2010.

- [6] L. G. Martínez, G. Licea, A. Rodríguez, J. R. Castro, O. Castillo, Using MatLab's fuzzy logic toolbox to create an application for RAMSET in software engineering courses. *Computer Applications in Engineering Education*, 21(4): 596–605, December 2013.
- [7] L. F. Capretz, Personality types in software engineering. *International Journal of Human-Computer Studies*, 58 (2): 207–214, 2003.
- [8] N. Salleh, E. Mendes, J. Grundy, Investigating the effects of personality traits on pair programming in a higher education setting through a family of experiments. *Empirical Software Engineering*, 19 (3):714–752, Jun 2014.
- [9] N. Gorla, Y. W. Lam, Who should work with whom? Building effective software project teams. *Communications of the ACM*, 47(6): 79–82, 2004.
- [10] J. Chao, G. Atli, Critical Personality Traits In Successful Pair Programming. In: *Proceedings of Agile 2006 Conference*, pp. 89–93, 2006.
- [11] A. D. D. Cunha, D. Greathead, Does personality matter? An analysis of code-review ability. *Communications of the ACM*, 50(5): 109–112, 2007.
- [12] P. Sfetsos, I. Stamelos, L. Angelis, I. Deligiannis, Investigating the impact of personality types on communication and collaboration-viability in pair programming – an empirical study. *Proceedings of the 7th International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2006)*, pp. 43–52, 2006.
- [13] G. J. Boyle, Myers-Briggs type indicator (MBTI): some psychometric limitations. *Australian Psychologist*, 30(1): 71–74, 1995.
- [14] K. S. Choi, A discovery and analysis of influencing factors of pair programming. Unpublished Ph.D. Dissertation, New Jersey Institute of Technology, USA 2004.
- [15] K. S. Choi, F. P. Deek, Im I., Exploring the underlying aspects of pair programming: the impact of personality. *Information and Software Technology*, 50(11): 1114–1126, 2008.
- [16] N. Salleh, E. Mendes, J. Grundy, Empirical studies of pair programming for CS/SE teaching in higher education: A systematic literature review. *IEEE Trans Software Eng*, 37(4): 509–525, 2011.
- [17] G. S. J. Burch, N. Anderson, Personality as a predictor of work-related behavior and performance: recent advances and directions for future research. In: *Hodgkinson G. P., Ford J. K. (eds) International review of industrial and organizational psychology*. Wiley UK, pp. 261–305, 2008.
- [18] M. R. Barrick, M. K. Mount, The big five personality dimensions and job performance: a meta-analysis. *Personality Psychology*, 44: 1–26, 1991.
- [19] P. Sfetsos, I. Stamelos, L. Angelis, I. Deligiannis, An experimental investigation of personality types impact on pair effectiveness in pair programming. *Empir. Softw. Eng.*, 14 (2): 187–226, 2009.
- [20] K. Lee, M. C. Ashton, Psychopathy, Machiavellianism, and narcissism in the Five-Factor Model and the HEXACO model of personality structure. *Personality Individ. Differ.*, 38 (7): 1571–1582, 2005.
- [21] R. R. McCrae, O. P. John, An introduction to the five-factor model and its applications. *Journal of Personality*, 60 (2): 175–215, 1992.
- [22] N. Schmitt, The interaction of neuroticism and gender and its impact on self-efficacy and performance. *Human Performance*, 21: 49–61, 2008.
- [23] J. E. Driskell, E. Salas, F. F. Goodwin, P. G. O'Shea, What makes a good team player? *Personality and team effectiveness. Group Dynamics: Theory, Research, and Practice*, 10(4): 249–271, 2006.
- [24] A. L. Pervin, *Psychologia osobowości*. GWP, Gdańsk 2002, s. 64.
- [25] M. A. Conard, Aptitude is not enough: how personality and behavior predict academic performance. *J Res Pers*, 40: 339–346, 2006.
- [26] K. Matzler, B. Renzl, J. Muller, S. Herting, T. A. Mooradian, Personality traits and knowledge sharing. *J Econ Psychol*, 29: 301–313, 2008.
- [27] B. De Raad, H. C. Schouwenburg, Personality in learning and education: a review. *Eur J Pers*, 10: 303–336, 1996.
- [28] N. Salleh, E. Mendes, J. Grundy, Investigating the effects of personality traits on pair programming in a higher education setting through a family of experiments. *Empir Software Eng*, 19: 714–752, 2014.
- [29] J. Karn, T. Cowling, A follow up study of the effect of personality on the performance of software engineering teams. In: *Proceedings of the 2006 ACM/IEEE International Symposium on Empirical Software Engineering*, ACM, pp. 232–241, 2006.
- [30] S. T. Acuña, M. Gómez, N. Juristo, How do personality, team processes and task characteristics relate to job satisfaction and software quality?, *Inf Softw.Technol.*, 51 (3): 627–639, 2009.
- [31] G. H. Grabowska, Bliskość emocjonalna w tworzeniu zespołów projektowych. *Zeszyty Naukowe Wydziału Informatycznych Technik Zarządzania Wyższej Szkoły Informatyki Stosowanej i Zarządzania „Współczesne Problemy Zarządzania”*, 1, 2013.
- [32] B. Zawadzki, J. Strelau, P. Szczepaniak, M. Śliwińska, *Inwentarz osobowości NEO-FFI Costy i McCrae*. Pracownia Testów Psychologicznych PTP, Warszawa 1998, s. 7–34.
- [33] E. Diener, E. M. Suh, R. E. Lucas, H. L. Smith, Subjective well-being: Three decades of progress, *Psychological Bulletin*, 125: 276–302, 1999.
- [34] J. Suls, P. Green, S. Hillis, Emotional reactivity to everyday problems, affective inertia, and neuroticism. *Personality and Social Psychology Bulletin*, 24: 127–136, 1998.
- [35] A. Lee, R. O. Pihl, Prefrontal cognitive ability, intelligence, Big Five personality and the prediction of advanced academic and workplace performance. *Journal of Personality and Social Psychology*, 93: 298–319, 2007.
- [36] S. Rothmann, E. P. Coetzer, The big five personality dimensions and job performance, *SA Journal of Industrial Psychology*, 29(1): 68–74, 2003.
- [37] M. Tamir, M. D. Robinson, Knowing good from bad: The paradox of neuroticism, negative affect, and evaluative processing. *Journal of Personality and Social Psychology*, 87: 913–935, 2004.
- [38] K. Horney, *Neurotyczne osobowości naszych czasów*. Rebis, Poznań 2011, s. 62–68.
- [39] J. F. Salgado, The big five personality dimensions and counterproductive behaviours. *International Journal of Selection and Assessment*, 10(1/2): 117–125, 2002.
- [40] C. G. Cegielski, D. J. Hall, What makes a good programmer? *Comm. ACM*, 49 (10): 73–75, Oct. 2006.
- [41] M. O. Laney, *The Introvert Advantage: How to Thrive in an Extrovert World*. NY: Workman Publishing, 2002.
- [42] J. M. Digman, N. K. Takemoto-Chock, Factors in the Natural Language of Personality: Re-analysis, Comparison, and Interpretation of Six Major Studies, *Multivariate Behavioral Research*, 16: 149–170, 1981.
- [43] R. Hogan, Socioanalytic theory of personality. In M. M. Page (Ed.), *1982 Nebraska Symposium on Motivation: Personality—current theory and research*, pp.55–89. Lincoln: University of Nebraska Press, 1983.
- [44] P. T. Jr. Costa, R. R. McCrae, Influence of extraversion and neuroticism on subjective well-being: Happy and unhappy people. *Journal of Personality and Social Psychology*, 38: 668–678, 1980.
- [45] M. K. Rothbart, J. E. Bates, Temperament. In W. Damon (Series Ed.), N. Eisenberg (Vol. Ed.), *Handbook of child psychology: Vol. 3. Social, emotional and personality development*, (5th Ed), New York: Wiley, 1998, pp. 105–176.
- [46] E. K. Gray, D. Watson, General and specific traits of personality and their relation to sleep and academic performance, *Journal of Personality*, 70: 177–206, 2002.
- [47] D. M. Higgins, J. B. Peterson, A. Lee, R. O. Pihl, Prefrontal cognitive ability, intelligence, Big Five personality and the prediction of advanced academic and workplace performance, *Journal of Personality and Social Psychology*, 93: 298–319, 2007.
- [48] S. Dewitt, H. C. Schouwenburg, Procrastination, temptations, and incentives: The struggle between the present and the future in procrastinators and the punctual. *European Journal of Personality*, 16 (6): 469–489, 2002.
- [49] M. K. Mount, M. R. Barrick, G. L. Stewart, Five-factor model of personality and performance in jobs involving interpersonal interactions. *Human Performance*, 11 (2): 145–165, 1998.