

BADANIE AKCEPTACJI OPROGRAMOWANIA OPEN SOURCE NA WYDZIAŁACH INŻYNIERSKICH UCZELNI TECHNICZNEJ

Sławomir RADOMSKI¹, Adam MUC², Adam SZELEZIŃSKI³, Piotr MYŚIAK²

1. Gdańska Szkoła Wyższa, Wydział Nauk Inżynierskich
e-mail: s.radom@wp.pl
2. Akademia Morska w Gdyni, Wydział Elektryczny
e-mail: a.muc@we.am.gdynia.pl, p.mysiak@we.am.gdynia.pl
3. Wydział Mechaniczny, Akademia Morska w Gdyni
e-mail: a.szelezinski@wm.am.gdynia.pl

Streszczenie: Masowa produkcja oprogramowania powoduje, że do wykonania określonych zadań można użyć programów od różnych producentów. Nie wszystkie jednak programy cieszą się taką samą popularnością. Badanie cech, które czynią, że dany program jest bardziej akceptowany niż inny, o podobnej funkcjonalności, stało się niezbędne i jest sukcesywnie rozwijane. W przypadku młodej kadry inżynierskiej przyzwyczajenia odnośnie używanego oprogramowania technicznego kształtowane są na uczelni, na której zdobywają wiedzę i umiejętności. Uczelnie techniczne bardzo często bazują na specjalistycznym oprogramowaniu w procesie kształcenia. Jest ono ważne zwłaszcza przy projektowaniu zajęć praktycznych, typu projekty lub laboratoria. Oprogramowanie specjalistyczne bywa często bardzo drogie, a przez to niedostępne dla studentów. Naturalną alternatywą dla tej sytuacji jest możliwość korzystania z oprogramowania *open source*.

Słowa kluczowe: akceptacja oprogramowania, model UTAUT, *open source*, inżynieria oprogramowania.

1. WSTĘP

Różnorodność języków programowania, dostępność środowisk programistycznych i inżynierii oprogramowania powoduje, że oprogramowanie może być produkowane nie tylko przez wyspecjalizowane firmy ale również przez zespoły ludzi pasjonujących się programowaniem i informatyką. W efekcie, na rynku oprogramowania dostępne są różne wersje programów lub nawet systemów informatycznych spełniających podobne zadania. Dlatego do wykonania określonego projektu można wykorzystać dowolny program o podobnej funkcjonalności. Może to być oprogramowanie mające charakter komercyjny (zamknięty), bądź otwarty (*ang. open source*). Oprogramowanie *open source* staje się coraz bardziej popularne w edukacji, administracji, ale też i technice. Duża liczba organizacji decyduje się na oprogramowanie *open source* w celu obniżenia kosztów użytkowania i zwiększenia dostępności oprogramowania. Czasami też, oprogramowanie *open source* charakteryzuje się większym bezpieczeństwem i niezawodnością niż porównywalne oprogramowanie komercyjne. Wśród odbiorców oprogramowania *open source* są też uczelnie wyższe, które wykorzystują je do wspomagania procesu kształcenia na technicznych kierunkach

studiów. Dlatego też ciekawym wydaje się zbadanie zakresu rozpowszechnienia tego typu oprogramowania na uczelni i wśród studentów.

2. AKCEPTACJA I ADOPCJA OPROGRAMOWANIA

Oprogramowanie, nawet to specjalistyczne, cechuje duża różnorodność. Stworzono wiele rodzajów oprogramowania dla tej samej klasy zastosowań, np. systemów operacyjnych, kompilatorów języków programowania, systemów oprogramowania biurowego, edukacyjnego, systemów klasy ERP, CRM, pakietów elektronicznego obiegu dokumentów WFM, systemów *Business Intelligence*, internetowych środowisk programistycznych czy nawet internetowych laboratoriów fizyki, elektroniki i innych. Wymienione rodzaje systemów oprogramowania mają charakter komercyjny (zamknięty), bądź otwarty. Prowadzone przez firmy doradcze i informatyczne rankingi wskazują, iż programy i systemy uzyskują wysoki stopień akceptacji w postaci wielkości sprzedaży, czy też zakresu rozpowszechnienia.

Świadectwem akceptacji oprogramowania jest to, czy jego użytkownicy, w swojej bieżącej działalności, będą je wykorzystywać w celu wykonywania swojej pracy zawodowej lub realizacji zainteresowań. A zatem, przez akceptację oprogramowania należy rozumieć podjęcie przez użytkownika danego programu (systemu) decyzji o zachowaniu tego właśnie programu (systemu), spośród dostępnego zbioru programów bądź systemów o identycznej lub zbliżonej funkcjonalności. W literaturze przedmiotu używa się też, różnego od akceptacji, pojęcia adopcji, które oznacza aprobatę użytkownika programu (systemu) w firmie lub instytucji, na podstawie akceptacji programu (systemu). A zatem, pojęcie akceptacji stosuje się w odniesieniu do indywidualnego użytkownika, natomiast pojęcie adopcji dotyczy organizacji. Akceptacja oprogramowania ma więc decydujący wpływ na jego adopcję przez firmy i instytucje, a zatem rozpowszechnienie jego implementacji.

Stąd zidentyfikowanie przyczyn zachowania czy zaniechania użytkownika określonego systemu oprogramowania jest ważną przesłanką dla podejmowania decyzji o zakupie lub wdrożeniu danego systemu w uczelni

czy innej instytucji. Konstatacja ta stała się decydującym powodem popularności metod badania akceptacji oprogramowania dla organizacji: gospodarczych, administracyjnych i edukacyjnych. Takie badania są szczególnie wspierane w krajach o wysokim poziomie zastosowań technologii teleinformatycznych. Również oprogramowanie o kodzie otwartym zyskuje w ostatnich latach na popularności. Jak wskazują badania wykonane przez Pentor [1] na zlecenie HP, Novell oraz Oracle, 19 na 20 polskich firm korzysta w różnym zakresie z oprogramowania *open source*. Około 94% firm używa otwartego systemu operacyjnego Linux na serwerach, a w 46% na stacjach roboczych. Jak wynika z tych badań trzy czwarte instytucji korzysta z otwartego oprogramowania serwerowego – głównie serwerów WWW i pocztowych oraz z programów typu ściana ognia. Jedna firma na pięć korzysta z *open source* w kluczowych zastosowaniach w swojej działalności. Raport Pentora przedstawia stwierdzenie, że aż dwie trzecie firm wykorzystuje systemy biznesowe klasy ERP i CRM z co najmniej częściowo otwartym kodem.

3. MODELE AKCEPTACJI OPROGRAMOWANIA

W celu wyznaczenia zmiennych objaśniających wpływających na akceptację technologii *open source* wykorzystano modele akceptacji oprogramowania (technologii) [2].

Jest duża grupa modeli pozwalających na badanie akceptacji oprogramowania przez użytkowników. Do najbardziej znanych modeli należą: TRA (ang. *Theory of Reasoned Action*), TAM/TAM2 (ang. *Technology Acceptance Model*), TPB (ang. *Theory of Planned Behavior*) czy UTAUT (ang. *Unified Theory of Acceptance and Use of Technology*). Są to modele oparte o teorię dyfuzji innowacji [3] lub akceptacji technologii [4, 5].

Inicjatorem badań akceptacji oprogramowania jest F. D. Davis. Najbardziej stymulujący wpływ na rozwój tej dziedziny badań miał jego artykuł [6]. Opracowany przez niego model TAM polegał na ocenie akceptacji oprogramowania przy uwzględnieniu dwu zmiennych objaśniających:

- **postrzegana łatwość użycia** (ang. *perceived easy of use*) – określona jako poziom odczucia danej osoby, że system (aplikacja) jest łatwa w użytkowaniu;
- **postrzegana użyteczność** (ang. *perceived usefulness*) – określona jako poziom odczucia danej osoby, że używanie systemu da jej korzyści w pracy (zwiększy jej efektywność, poprawi jej pozycję w firmie i środowisku).

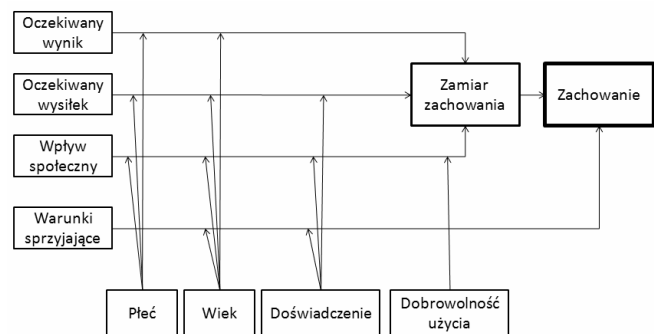
Obie zmienne mają decydujący wpływ na inicjowanie, umacnianie i kształtowanie zamiaru zachowania. Użytkownik oprogramowania, przez zamiar skorzystania z niego, wyraża finalną decyzję o zachowaniu danego rodzaju lub typu oprogramowania, po rozważeniu i ocenie warunków sprzyjających dla przejścia z zamiaru zachowania na zachowanie. Badania te F. D. Davis kontynuował i rozwijał wraz ze współpracownikami w kolejnych latach [6]. Cieszyły się one dużym zainteresowaniem i zostały podjęte przez innych naukowców. Stworzone zostały modele: TPB (ang. *Theory of Planned Behavior*) [7], czy UTAUT (ang. *Unified Theory of Acceptance and Use of Technology*) [8]. Właśnie model UTAUT i jego modyfikacja stał się przedmiotem zainteresowania w niniejszej pracy.

Czerpie on z wielu wcześniejszych modeli stosowanych w celu badania akceptacji oprogramowania

[9, 10]. UTAUT jest uważany za najlepszy model do analizy akceptacji i należy się spodziewać, że przyszłe badania będą nakierowane właśnie na użycie tego modelu. Schemat modelu UTAUT demonstruje rysunek 1.

W modelu UTAUT znaczący wpływ na zachowanie (pośrednio przez zamiar zachowania) mają cztery zmienne objaśniające:

- **oczekiwany wynik** (ang. *performance expectancy*) – definiowany jako stopień odczucia użytkownika, że wykorzystywanie danego oprogramowania pomoże mu osiągnąć dobre wyniki i korzyści w pracy,
- **oczekiwany wysiłek** (ang. *effort expectancy*) – definiowany jako stopień odczucia użytkownika, że oprogramowanie jest łatwe do opanowania,
- **wpływ społeczny** (ang. *social influence*) – definiowany jako stopień odczucia użytkownika, że ważne dla niego osoby zachęcają go do użycia danego oprogramowania,
- **warunki sprzyjające** (ang. *facilitating conditions*) – definiowane jako stopień odczucia użytkownika, że jego firma, wspomaga użycie danego oprogramowania.



Rys.1. Model UTAUT [8]

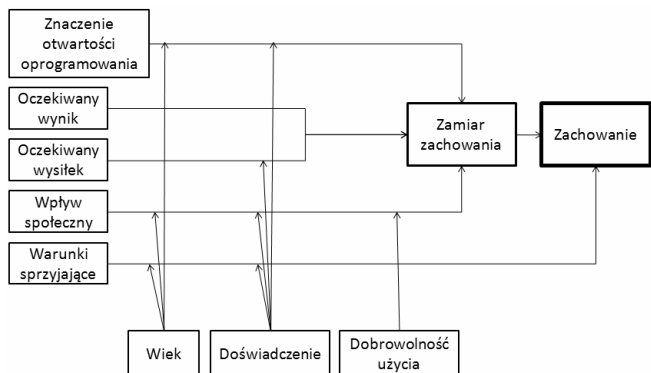
W modelu tym V. Venkatesh i inni wprowadzili także cztery zmienne, które pełnią rolę kluczowych moderatorów (ang. *key moderators*) i wywierają wpływ na powyższe zmienne niezależne. Moderatorami tymi są: płeć (ang. *gender*), wiek (ang. *age*), doświadczenie (ang. *experience*) i dobrowolność użycia (ang. *voluntariness of use*). W oparciu o powyższy model postawiono szereg hipotez, które potwierdził w swoich badaniach V. Venkatesh w roku 2003. Relacje zaznaczone na rysunku 1 mają postać funkcji liniowych, które analizowano za pomocą korelacji, a także metody PLS (ang. *Partial Least Squares*). Badanie akceptacji oprogramowania w tym zastosowaniu modelu UTAUT dotyczyły w głównej mierze oprogramowania komercyjnego, a w bardzo ograniczonym zakresie oprogramowania otwartego. W związku z tym w niniejszym badaniu postanowiono wykorzystać zmodyfikowany model UTAUT [11].

W strukturze modelu UTAUT dołączono piątą zmienną – znaczenie otwartości oprogramowania - ZOO (zmienną ukrytą). Zawiera ona w sobie zmienne obserwowalne bezpośrednio związane z oprogramowaniem *open source*, które nie są zawarte w pozostałych zmiennych ukrytych:

- niskie koszty licencji,
- możliwość modyfikacji kodu programu,
- stabilność rynku (związaną z brakiem uzależnienia od jednej firmy),
- większe bezpieczeństwo (związaną z możliwością modyfikacji kodu),
- wsparcie techniczne.

Zmienna ZOO zdefiniowana jako stopień odczucia użytkownika, który oznacza otwartość oprogramowania wyrażoną przez niskie koszty, bezpieczeństwo, otwartość kodu, niezależność od producenta, wsparcie techniczne ma wpływ na zamiar użycia oprogramowania (zamiar zachowania).

Z wcześniejszych badań wynika znikomy wpływ płci na zamiar zachowania, dlatego postanowiono zrezygnować z tej zmiennej modelu. Zmodyfikowany model UTAUT przedstawia rysunek 2.



Rys. 2. Zmodyfikowany model UTAUT

Najnowsze badania nakierowane są na analizę kolejnych czynników oraz ich wpływu na zamiar zachowania i zachowanie. Venkatesh, V., Thong, J., & Xu, X. w roku 2012 dołączyli do modelu UTAUT zmienne: motywacji (opartej na przyjemnym odczuciu z użytkowania oprogramowania ang. *Hedonic Motivation*), ceny (ang. *Price Value*) oraz doświadczenia i przyzwyczajenia (ang. *Experiance and Habit*). Nowopowstały model nazwano UTAUT 2 [12]. Badania potwierdzające zasadność modelu UTAUT przeprowadzono także w roku 2013 przez Taiwo, A., i Downe, A. Co ciekawe, badania zostały przeprowadzone na bardzo dużej próbie liczącej 11 000 użytkowników [13]. Przytoczone powyżej badania wskazują kierunek kolejnych analiz związanych z akceptacją oprogramowania oraz modelem UTAUT.

4. MODELE RÓWNAŃ STRUKTURALNYCH SEM

Jedną z technik, którą wykorzystuje się do badania modeli akceptacji oprogramowania są modele równań strukturalnych, a szczególnie częściowa metoda najmniejszych kwadratów (PLS).

Modelowanie równań strukturalnych SEM [14]. Dzięki takiej strukturze SEM, możemy budować modele, w których występują pośrednie oraz bezpośrednie powiązania między zmiennymi obserwowalnymi i zmiennymi nieobserwowalnymi bezpośrednio. Główną charakterystyką modeli SEM jest to, iż modele te uwzględniają występowanie błędów pomiarowych we wszystkich zmiennych [15]. W SEM przedstawiają liniowe zależności o charakterze przyczynowym. Podstawą wyznaczania parametrów modeli SEM są macierze kowariancji lub korelacji, które to określają powiązania wszystkich par obserwowalnych zmiennych. Do podstawowych technik modeli SEM zalicza się:

- analizę ścieżki (ang. *path analysis*),
- konfirmacyjną analizę czynnikową (ang. *confirmatory factor analysis*),

- modelowanie regresji strukturalnych (ang. *structural regression modelling*).

Klasyczna metoda wyznaczania parametrów modeli równań strukturalnych SEM wykorzystuje macierze wariancji i kowariancji między zmiennymi obserwowalnymi i ukrytymi. Dodatkowym ograniczeniem dla badań jest to, iż zmienne obserwowalne muszą łącznie mieć wielowymiarowy rozkład normalny, co w niektórych badaniach jest rzeczą trudną do uzyskania.

Zaletą metody PLS jest to, iż nie wymaga ona ograniczenia związanego z rozkładem zmiennych obserwowalnych, a także założeń związanych z rozkładem reszt czy z niezależnością obserwacji. Różnice między klasycznymi metodami SEM a metodą PLS przejawiają się także w minimalnej liczbie próby. Modele PLS pozwalają na określenie parametrów na bardzo małych próbach, czego nie można było dokonać na klasycznych modelach SEM. Następną zaletą, z punktu widzenia akceptacji oprogramowania, jest różnica w sposobie identyfikacji zmiennych ukrytych. Modele SEM dopuszczają istnienie relacji między zmiennymi ukrytymi i obserwowalnymi tylko w podejściu refleksyjnym (zmienna rzeczywiście istnieje), natomiast model PLS uwzględnia także relacje w podejściu formacyjnym (zmienna jest hipotetyczna). Dodatkową zaletą modelu PLS jest to, iż mogą one być duże, zaś klasyczne modele SEM muszą być małe lub co najwyżej średniej wielkości. Wszystkie te różnice między klasycznymi modelami SEM i modelem PLS wpływają na to, że częściej metoda najmniejszych kwadratów jest idealnym narzędziem do badania adopcji i akceptacji oprogramowania. Na świecie, szczególnie w Stanach Zjednoczonych oraz w krajach azjatyckich, przeprowadza się bardzo często badania z wykorzystaniem równań strukturalnych oraz metody PLS. Również w badaniu akceptacji oprogramowania metoda PLS sprawdza się doskonale [16, 17].

Podobnie jak w przypadku liniowej regresji wielorakiej, głównym celem regresji metodą najmniejszych kwadratów jest budowa modelu liniowego w postaci równania 1.

$$Y = X \cdot B + E \quad (1)$$

gdzie:

Y - oznacza macierz odpowiedzi o wymiarach n (liczba przypadków) na m (liczba zmiennych), X - oznacza macierz zmiennych objaśniających o wymiarach n (liczba przypadków) na p (liczba zmiennych), B - oznacza macierz współczynników regresji o wymiarach p na m , E - oznacza składnik losowy modelu o takich samych wymiarach jak macierz Y .

Do analizy wyników wykorzystano częściową metodę najmniejszych kwadratów (PLS). Metodę tą wybrano z dwóch powodów. Po pierwsze model miał strukturę formacyjną, czego nie można było wykonać za pomocą klasycznych modeli SEM, a po drugie częściowa metoda najmniejszych kwadratów PLS jest bardziej odpowiednia w analizie danych, kiedy model jest nowym modelem i nie był intensywnie testowany wcześniej. Stwierdzono także, że w literaturze dość rzadko spotyka się badania, w których wykorzystuje się powyższe metody, dlatego postanowiono właśnie za pomocą równań strukturalnych SEM analizować ten model [16, 17].

5. WYNIKI BADAŃ

Cechą przestrzenną użytkowników oprogramowania *open source* w badaniu, były wydziały inżynierskie na uczelni technicznej. Badanie obejmowało wybranych studentów z uczelni technicznej studiujących na kierunku elektrotechnika i mechanika, w którym wykorzystuje się otwarte oprogramowanie. Studenci wypełnili kwestionariusz ankietowy w pierwszym półroczu 2016 roku. Do badania wykorzystano 42 poprawnie wypełnione kwestionariusze ankietowe. Analizie poddano ilościowe cechy zmienne użytkowników oprogramowania *open source* takie jak: rzeczową, przestrzenną i czasową. Do analizy danych wykorzystano oprogramowanie SmartPLS.

Kwestionariusz ankietowy składał się z 15 pytań. Poza częścią identyfikującą zawierał on 7 pytań objaśniających poszczególne zmienne ukryte: zamiar zachowania (ZZ), oczekiwany wynik (OWYN), oczekiwany wysiłek (OWYS), wpływ społeczny (WPS), warunki sprzyjające (WAS), dobrowolność użycia (DOBU), doświadczenie, wiek. Zmienna ukryta znaczenie otwartości oprogramowania (ZOO) była wyznaczona poprzez uśrednienie otrzymanych odpowiedzi na zmienne obserwowalne charakteryzujące ZOO. Respondenci zaznaczali odpowiedzi w tych pytaniach według siedmiostopniowej skali. W dalszej części studium, za pomocą programu SmartPLS, wyznaczono wartości współczynnika całkowitego wpływu (T) zmiennych objaśniających na zmienne objaśniane.

Tabela 1. Wpływ zmiennych objaśniających na objaśniane

Zmienna objaśniana: ZAMIAR ZACHOWANIA		
	T	t
ZOO	0,35	4,05
OWYN	0,60	5,79
OWYS	0,30	3,00
WPS	0,29	2,99
Zmienna objaśniana: ZACHOWANIE		
ZZ	0,51	5,78
WAS	0,29	2,46
Zmienna objaśniana: ZNACZENIE OTWARTOŚCI OPROGRAMOWANIA		
WIEK	0,37	3,14
DOŚWIADCZENIE	0,35	3,92
Zmienna objaśniana: OCZEKIWANY WYSIŁEK		
DOŚWIADCZENIE	-0,45	3,01
Zmienna objaśniana: WPŁYW SPOŁECZNY		
WIEK	0,55	5,77
DOŚWIADCZENIE	0,35	3,48
DOBU	-0,54	5,71
Zmienna objaśniana: WARUNKI SPRZYJAJĄCE		
WIEK	0,31	3,37
DOŚWIADCZENIE	0,45	3,24

Współczynnik T uzyskuje wartości w przedziale [-1,1]. Interpretacja wartości współczynnika T przedstawia się analogicznie jak interpretacja współczynnika korelacji liniowej Pearsona - im wynik bliższy modułu jedności tym

silniejszy związek między zmiennymi, a im bliżej zera tym słabszy związek między zmiennymi. Poprzez wykorzystanie dołączonego do programu modułu *bootstrap* wyznaczono także wartości statystyki t-Studenta (t). Wyniki uzyskane w badaniu przedstawia tabela 1. Moduł *bootstrap* wykorzystuje metodę szacowania rozkładu błędów estymacji, za pomocą wielokrotnego losowania ze zwracaniem z próby. Metoda ta jest przydatna szczególnie wtedy, gdy nie jest znana postać rozkładu zmiennej w populacji. Dla statystyki t-Studenta, przy założonym błędzie 5% ($p = 0,05$) dla $n=42$ wartość krytyczna testu t-Studenta wyniosła 2,018. Ponieważ wyniki testu t-Studenta w powyższym badaniu wyniosły ponad tę wartość, oznacza to, że uzyskane wyniki są istotne statystycznie dla poziomu istotności $p = 0,05$. Związki nieistotne statystycznie zostały usunięte z wyników badań.

5. WNIOSKI KOŃCOWE

Badania wykazały bardzo silny związek oczekiwanego wyniku i zamiaru zachowania. Wynik ten należy interpretować w ten sposób, iż funkcjonalność oprogramowania *open source* wpływa bardzo silnie na jego wybór. Równie ważne jak czynnik funkcjonalności programu okazuje się to, czy za jego pomocą użytkownik będzie mógł wykonać zadanie szybciej i czy będzie bardziej produktywny. Ankietowani odpowiadając na pytania przypisane do zmiennej oczekiwany wynik w sposób pośredni potwierdzili przydatność oprogramowania *open source* na uczelni. Równie silny związek zaobserwowano także w relacji zamiaru zachowania i zachowania. Badania te udowodniły stwierdzenie leżące u podstaw większości teorii akceptacji oprogramowania, że jeżeli użytkownik ma zamiar użycia danego typu programu to go użyje. Wysoki stopień korelacji uzyskały także relacje wpływu wieku na zmienną wpływ społeczny co świadczy o tym, że im starszy student tym bardziej bierze pod uwagę głos starszego kolegi, czy wykładowcy oraz wpływ dobrowolności na wpływ społeczny, z tym, że w tym przypadku uzyskano ujemną wartość współczynnika T. Ujemna wartość współczynnika przedstawia zależność w drugą stronę, czyli dobrowolność użycia tego programu przez studenta jest mocno ograniczona przez wpływ osób ściśle powiązanych ze studentem np. prowadzących zajęcia. W relacji doświadczenie oczekiwany wysiłek (DOŚW – OWYN) także uzyskał ujemną wartość współczynnika T, co pokazuje, że im student ma większe doświadczenie w używaniu konkretnego typu oprogramowania, tym jego oczekiwania co do wysiłku jaki będzie musiał podjąć w celu wykonania zadania są mniejsze.

Statystycznie istotne okazały się także pozostałe relacje modelu, w których zmienną objaśnianą jest zamiar zachowania.

Badania wskazały także, iż oczekiwany wysiłek także ma istotny wpływ na zamiar zachowania. Świadczy to o tym, że studenci wybierają chętniej programy komputerowe, które albo są proste w użyciu, albo nie wymagają zbyt dużej ilości czasu na naukę. Zaobserwowano więc, że łatwość opanowania danego programu *open source* wpływa także na jego wybór.

Kolejną zmienną objaśnianą mającą wpływ na zamiar zachowania jest wpływ społeczny. Często inne ważne dla użytkownika osoby, takie jak profesor lub inny nauczyciel akademicki, ale także koledzy, znajomi, lub inne osoby, w jakikolwiek sposób z nim związane, mają bezpośredni wpływ na wybór oprogramowania. Udowodniono także wpływ warunków sprzyjających na zachowanie.

Zmienna objaśniająca warunki sprzyjające zawiera w sobie informację, czy użytkownik ma niezbędne umiejętności i wiedzę teoretyczną do używania oprogramowania *open source*. Udowodniono więc, że im student, który już wcześniej posiadał umiejętność używania tego oprogramowania wybierze je chętniej w dalszej swojej edukacji. Zbadano również wpływ zmiennych kluczowych (wieku, doświadczenia, dobrowolności użycia) na znaczenie otwartości oprogramowania. Okazało się, że wiek i doświadczenie wpływają znacząco na znaczenie otwartości oprogramowania. Im student starszy i ma większe doświadczenie w użytkowaniu konkretnego oprogramowania to uważa, że znaczenie ma darmowość, oraz możliwość analizy kodu programu. Przeprowadzone i zaprezentowane badania wykazały możliwość identyfikacji czynników akceptacji oprogramowania *open source* przy pomocy zmodyfikowanego modelu UTAUT.

6. BIBLIOGRAFIA

1. Pentor: Linux i Open Source w biznesie, III Kongres Linuxa profesjonalnego, 2012
2. Wang W., Benbasat I.: Trust in and Adoption of Online Recommendation Agents, "Journal of the Association for Information Systems", 6(3), 2005
3. Moore G.C., Benbasat I.: Development of an Instrument to Measure the Perceptions of Adopting an Information Technology Innovation, *Information Systems research* (2:3), 1991, s. 192-222
4. Davis F.D., Bagozzi R.P., Warshaw P.R.: User acceptance of computer technology: A comparison of two theoretical models, "Management Science" 1989, 35(8), s. 982-1003
5. Davis F., Bagozzi R., Warshaw P.: Extrinsic and Intrinsic Motivation to Use Computers in the Workplace, *Journal of Applied Social Psychology*, 22:14, 1992, s. 1111-1132
6. Davis F.: Perceived usefulness, perceived ease of use, and user acceptance of information technology, *MIS Quarterly*, 13(3), 1989, s. 319-339
7. Ajzen I.: The theory of planned behavior, *Organizational Behavior and Human Decision Processes*, 50(2), 1991, s. 179-211
8. Venkatesh V., Morris M.G., Davis G., Davis F.: User acceptance of information technology: Toward a unified view, *MIS Quarterly*, 27(3), 2003, s. 425-478
9. Zhu K., Kraemer K.L., Gurbaxani V., Xu S.: Migration to Open-Standards Interorganizational Systems: Network Effects, Switching Costs and Path Dependency, *Ciro Consortium Research Report*, 2005
10. Zhu K., Kraemer K.L., Xu S.: The Process of E-Business Assimilation in Organizations: A Technology Diffusion Perspective, "Management Science", Vol. 52, No. 10, 2006
11. Radomski S.: Badanie akceptacji oprogramowania open source z wykorzystaniem zmodyfikowanego modelu UTAUT, rozprawa doktorska, 2011
12. Venkatesh, V., Thong, J., Xu, X.: Consumer Acceptance and Use of Information Technology: Extending the Unified Theory of Acceptance and Use of Technology. *MIS Quarterly*, 36(1), 2012, s. 157-178
13. Taiwo A., Downe A.: (The Theory Of User Acceptance And Use Of Technology (UTAUT): A Meta-Analytic Review Of Empirical Findings. *Journal of Theoretical and Applied Information Technology*, 49(1), 2013, s. 48-58
14. Gatnar E.: Statystyczne modele struktury przyczynowej zjawisk ekonomicznych, *Prace Naukowe Akademii Ekonomicznej*, Katowice, 2003
15. Strzała K.: Modelowanie równań strukturalnych: koncepcja i podstawowe pojęcia, „Prace i Materiały Wydziału Zarządzania”, nr 2, Sopot, 2006
16. Teo H.H., Wei K.K., Benbasat I.: Predicting Intention to Adopt Interorganizational Linkages: An Institutional Perspective, *MIS Quarterly*, (27:1), 2003
17. Tornatzky L., Fleischer M.: *The Process of Technological Innovation*, Lexington, MA: Lexington Books, 1990

TESTING OF OPEN SOURCE SOFTWARE ACCEPTANCE ON THE ENGINEERING FACULTIES OF UNIVERSITIES OF TECHNOLOGY

The massive production of software contributes to the possibility of using the programmes from different producers in order to complete definite tasks. Not all of the programmes, however, are so popular. The testing of software features about the similar functionality shows that some software are more acceptable than the other. This is required and is successfully developed. The habits and preferences of young engineers staff connected with technical software are shaped during studies at the university where they acquire knowledge and abilities. Universities of technology very often base their classes on special software during the process of education. It is especially important while designing a draft of practical classes such as laboratories and projects. Nevertheless, this specialized software is very expensive and at the same time unavailable for students. A natural alternative for this situation is the possibility to use open source software.

Keywords: software acceptance, UTAUT model, open source, software engineering.

