

# NAUCZANIE PRZEZ DZIAŁANIE W PRAKTYCE. PROGRAMOWANIE ZESPOŁOWE OCZAMI OPIEKUNA ZESPOŁU.

Maciej Koziński

Wydział Matematyki i Informatyki, UMK w Toruniu

ul. Chopina 12/18, 87-100 Toruń

[mkozinsk@mat.umk.pl](mailto:mkozinsk@mat.umk.pl)

*Abstract. The Collective Programming is a good example of learning by doing. Since 1999 in our Faculty, these activities help our students to develop programming, electronic engineering and math skills as well as such "soft" skills like creativity and brainstorming, planning, marketing, communication. The final competition helps to motivate teams, evaluates the value of projects and – for the winners – gets a boost increase in self-confidence.*

## 1. Wstęp

Programowanie Zespołowe jest dobrym przykładem praktyki *nauczania przez działanie*. Obecnie na naszym Wydziale od 1999 roku, te zajęcia pomagają naszym studentom rozwijać "twarde" umiejętności zawodowe w zakresie programowania, projektowania układów elektronicznych czy stosowania narzędzi matematycznych, jak również "miękkie", takie jak kreatywność (burze mózgów), planowanie, marketing (planowanie produktu) czy komunikacja.

W trakcie Programowania Zespołowego studenci z własnej inicjatywy poznają technologie, których jeszcze nie poznali w ramach zajęć. Dotyczy to zarówno języków programowania czy narzędzi projektowych, jak i bibliotek programistycznych, *frameworks* czy publicznych interfejsów programistycznych (API) do usług (np. Google czy Facebooka, ale i licznych innych).

## 2. Formalny przebieg przedmiotu

Programowanie Zespołowe ma miejsce na III roku I stopnia studiów informatycznych, zarówno licencjackich, jak i inżynierskich. Przedmiot zazwyczaj jest inicjowany w trzeciej dekadzie października i zaczyna się od spotkania koordynatorów z wszystkimi studentami biorącymi udział w Programowaniu Zespołowym, podzie-

lenia studentów na 4-6 osobowe zespoły projektowe oraz przydzielenia poszczególnym zespołom opiekunów.

Po wyznaczeniu opiekunów, studenci są zobowiązani niezwłocznie skontaktować się z opiekunem. Wybierany jest studencki kierownik zespołu, sekretarz oraz – jeśli zachodzi taka potrzeba – wyznaczane są inne osoby na niezbędne funkcje. Wybierany jest temat projektu, chyba, że temat ten jest zlecany przez opiekuna. Zespół zakłada stronę WWW projektu na przeznaczonym do tego serwerze, gdzie opisuje projekt oraz publikuje notatki na temat spotkań zespołu i opiekuna.

Pod koniec semestru zimowego członkowie zespołu składają koordynatorom przedmiotu raport z dotychczasowych działań. W pierwszych dniach semestru letniego następuje ostateczny termin zmian funkcji w zespole.

Krótko przed prezentacjami zespół zamyka projekt i udostępnia na stronie WWW projektu jego dokumentację. Opiekunowie deklarują czy dopuszczają zespół do publicznej prezentacji.

Prezentacja odbywa się zazwyczaj w drugim tygodniu maja, w czwartek. W tygodniu poprzedzającym koordynatorzy, w obecności kierowników zespołów, losują kolejność prezentacji. W dni poprzedzające dzień prezentacji – zazwyczaj wtorek i środę – zespoły sprawdzają i ćwiczą przeprowadzanie prezentacji w auli. Próby obejmują część techniczną – podłączenie komputerów do rzutników i nagłośnienia, podłączenie dodatkowych urządzeń – oraz właściwy pokaz prezentacji.

W dniu prezentacji zespoły mają po 20-25 minut na zaprezentowanie projektu. Pięć minut (a czasem nieco więcej – jeśli zespół nie wykorzysta czasu przeznaczonego na prezentację) jest przeznaczone na zadawanie pytań przez jury i publiczność (prezentacja jest otwarta dla osób z zewnątrz oraz transmitowana *online* przez internetową Telewizję UMK – <http://tv.umk.pl/>). W skład jury wchodzi koordynatorzy przedmiotu i opiekunowie zespołów.

Po zakończeniu prezentacji jury udaje się na obrady, w trakcie których decyduje o zaliczeniu przedmiotu przez studentów oraz o przyznaniu nagród Dziekana Wydziału Matematyki i Informatyki. Wyniki konkursu ogłaszane są niezwłocznie po zakończeniu obrad jury.

Sformalizowany opis przedmiotu w niewielkim stopniu oddaje całe bogactwo działań, jakie mają miejsce zarówno w pracach zespołów, jak i poza zespołem.

### 3. Zespół

Zespoły najczęściej powstają spontanicznie, poprzez dobór w grupach koleżeńskich jeszcze podczas przedmiotów poprzedzających Programowanie Zespołowe. Taki scenariusz nie zawsze jest możliwy. Niekiedy pozostaje kilkoro studentów, którzy nie znaleźli się w zespołach powstających spontanicznie i koordynatorzy tworzą osobny zespół lub przydzielają pozostałych studentów do już istniejących

zespołów. W drugiej z wymienionych sytuacji wszyscy zainteresowani powinni podjąć starania na rzecz zintegrowania nowego członka zespołu i spożytkowania jego zdolności.

Jak każda grupa społeczna, zespół posiada swoją hierarchię społeczną. Biorąc pod uwagę efektywność pracy nad projektem najlepiej jest, jeśli w zespole jest wyraźny lider (kierownik), dwie lub trzy osoby o dużej inwencji i umiejętności rozwiązywania problemów (*inventor/troubleshooter*) i dwie lub trzy osoby o bardziej introwertycznym charakterze i skłonności do uporządkowanej, przewidywalnej i zaplanowanej przez innych pracy. Skład taki najlepiej pasuje do charakteru prac: projekt zazwyczaj ma charakter rozwojowy, niezbędne jest podejmowanie decyzji o organizacji bieżących prac (kto i w jaki sposób ma pracować nad częścią projektu czy rozwiązaniem konkretnego problemu), jak i decyzji technicznych mniejszej wagi (np. czy konfigurację oprogramowania zapisać w pliku czy w dodatkowej tabeli bazy danych), rozwiązywanie problemów czy wymyślanie rozwiązań technicznych (np. w jaki sposób zapewnić komunikację pomiędzy użytkownikami aplikacji) oraz prowadzenie łatwiejszych, przewidywalnych, pracochłonnych, zaplanowanych prac (np. przygotowanie interfejsu użytkownika, przygotowanie ekranu konfiguracyjnego aplikacji). Pracujący nad projektem zespół potrzebuje ludzi o każdym usposobieniu.

Zarówno nadmiar, jak i niedobór osób aktywnych i twórczych może powodować trudności w podejmowaniu decyzji osiaganiu porozumienia. Rolą opiekuna jest wyjaśnienie studentom konieczności podejmowania sprawnych decyzji i komunikacji w zespole, łagodzenie ewentualnych napięć czy motywowanie do podejmowania decyzji – zależnie od sytuacji i nastrojów panujących w zespole.

Zespół – bez działu opiekuna – może zgłosić koordynatorom postulat wyłączenia ze składu zespołu osoby, która nie współpracuje.

## 4. Opiekunowie

Opiekunem zespołu studenckiego może być zarówno pracownik naukowy lub techniczny Wydziału Matematyki i Informatyki, innej jednostki Uniwersytetu Mikołaja Kopernika w Toruniu lub zewnętrznej instytucji lub firmy, która współpracuje z Wydziałem w ramach Programowania Zespołowego.

Przez kilka lat zespołami zajmowały się również pary opiekun – pracownik Uniwersytetu – i druga osoba, będąca konsultantem zewnętrznym – informatycznym lub merytorycznym.

Głównym zadaniem opiekuna jest kształtowanie całego przebiegu projektu, od planowania, poprzez korygowanie przebiegu prac, po przygotowanie studentów do finałowej prezentacji.

Rola opiekuna jest specyficzna. Nie jest on typowym wykładowcą czy prowadzącym ćwiczenia. W zakresie przekazywania wiedzy – a raczej dzielenia się wie-

dzą i doświadczeniem – funkcjonuje na pograniczu obu tych form. Jednak rola opiekuna wykracza poza te formy zajęć. Jest on także doradcą, mentorem, osobą motywującą. Jeśli zespół nie potrafi podjąć decyzji w ważnych sprawach dotyczących projektu, wątpliwości rozstrzyga opiekun.

Choć opiekun pełni rolę mentora i doradcy, to rola ta – moim zdaniem – nie powinna być zbyt inwazyjna. Ważne jest pozostawienie studentom inicjatywy a nawet zachęcanie do podejmowania działań, wyzwań i wdrażania własnych pomysłów i brania za nie odpowiedzialności. W ten sposób również mogą odczuć skutki własnych działań i zaniechań – pozytywne, jak i negatywne. Opiekun powinien kształtować świadomość studentów w zakresie konsekwencji podejmowanych decyzji.

## 5. Projekt

Bardzo często powstały spontanicznie zespół studencki wybiera temat projektu jeszcze przed spotkaniem z opiekunem. Zazwyczaj rolą opiekuna jest wtedy dostosowanie projektu do przedmiotu. Wchodzą tu w grę okoliczności takie jak dostosowanie pracochłonności projektu do czasu trwania przedmiotu – projekt nie może być zbyt ambitny, gdyż będzie niewykonalny w założonym czasie, ale nie powinien też być zbyt mało wymagający. Również w ocenie innych aspektów projektu opiekun powinien naprowadzić studentów na sformułowanie założeń, w szczególności:

- celowości: choć projekt nie jest przygotowywany do sprzedaży, to jednak powinien posiadać cechy marketingowe dobrego produktu: zapewniać zaspokojenie potrzeb, być skierowanym do określonej grupy docelowej, wypełniać lukę rynkową,
- sprawdzenia, czy bliźniaczo podobny projekt już nie istnieje; jeżeli miałyby mieć miejsce stworzenie alternatywnej aplikacji, to podstawowe pytanie brzmi: co możemy zrobić lepiej od twórców już istniejących produktów? czy w czasie, gdy my będziemy rozwijać nasz projekt, oni nie rozwiną swojego bardziej?
- niezależności projektu od osób i zasobów spoza zespołu i Wydziału; stworzenie nowego portalu społecznościowego – to brzmi dumnie – ale kto zapełni ten portal treścią?

Staranne przygotowanie wizji projektu jest podstawą sukcesu. Konieczne jest odpowiedzenie sobie na pytania:

- co dokładnie robimy?
- jakie funkcjonalności ma to mieć?
- dla jakiego odbiorcy?
- w jaki sposób odbiorca będzie tego używał?

- jak to ma wyglądać?
- jakimi środkami to osiągniemy?

Bez starannego przygotowania i szczegółowego opisu projektu trudno jest uporządkować prace, zwłaszcza, jeśli odbywają się one kilka miesięcy po początkowej dyskusji.

Koniecznością w projekcie jest współpraca wewnątrz zespołu studenckiego w ramach kontaktu bezpośredniego (spotkania studentów i opiekunów) i elektronicznego (poczta elektroniczna, ale coraz częściej dedykowane grupy na Facebooku), wymiany dokumentów i plików (usługi w chmurze Google Drive, Dropbox) oraz współdzielenie i wersjonowanie kodu (SVN, GIT).

Coraz częściej jednak studenci sięgają do osób z zewnątrz, zadając konkretne pytania twórcom używanych rozwiązań (jak w przypadku tegorocznego projektu BICOM – pytania do twórców narzędzia do rozpoznawania mowy CMU Sphinx) oraz potencjalnym użytkownikom.

Dobrze jest, jeśli studenci notują na bieżąco zadania do wykonania, pomysły, spostrzeżenia. Niestety, nie wszyscy mają taki nawyk. Należy go kształtować, gdyż np. zgubienie odnośnika do strony zawierającej opis rozwiązania jakiegoś problemu skutkuje później stratą czasu na ponowne odszukanie tego zasobu.

Nie istnieje formalny wymóg opanowania jakiegokolwiek materiału – nie można mówić o podstawie programowej. Jednak zdobycie wiedzy i umiejętności technicznych, merytorycznych i społecznych jest niezbędne do zrealizowania projektu. Realizując projekt studenci mogą – i powinni – korzystać z już istniejących technologii i rozwiązań w zakresie np. interfejsu graficznego użytkownika, baz danych, komunikacji sieciowej – chyba, że cele ich pracy wymagają np. rozszerzenia istniejących rozwiązań lub odpowiednie rozwiązania nie istnieją. Niezbędne jest biegłe poruszanie się w dziedzinie projektu – pozytywnym przykładem może być tu opisany w dalszej części projekt Sim Cor. Zasięganie wiedzy u innych osób – twórców użytkowanych narzędzi czy końcowych użytkowników – wymaga rozwinięcia kompetencji społecznych, przede wszystkim w zakresie komunikacji i empatii. Publiczne zaprezentowanie projektu wymaga opanowania elementów przekazu werbalnego i niewerbalnego, a także zapanowania nad treścią i możliwymi do przewidzenia sytuacjami awaryjnymi (przygotowanie scenariusza działań).

Konkursowy charakter przedmiotu jest czynnikiem najbardziej motywującym studentów do pracy. Wyzwanie, rywalizacja między zespołami i potencjalny prestiż z końcowego wyróżnienia, jak i swobodny charakter prac i możliwość wyboru projektu i środków do jego realizacji nadają przedmiotowi charakter przygody. Jeżeli zespół od początku uświadamia sobie ten stan (albo studentom uświadomi to opiekun), to jest to dobra podstawa motywacyjna przez cały czas trwania projektu. Ze-

spółowy charakter i wspólne podjęcie wyzwania ułatwiają utrzymanie wysokiej mobilizacji.

## 6. Prezentacja

Końcowym przedsięwzięciem jest zaprezentowanie projektu, jego celów, osiągnięć i przebiegu prac jury i publiczności. Jest to najlepszy test kompetencji studentów, nieraz drastycznie obnaża braki projektu, jak i niedopatrzenia i zaniedbania zespołu. Jednocześnie mobilizuje zespoły do dopracowania projektów i usuwania usterek.

Przygotowanie prezentacji rozpoczyna się od scenariusza:

- co chcemy powiedzieć?
- w jaki sposób?
- jakich środków wyrazu chcemy użyć?

Równocześnie z ćwiczeniem i doskonaleniem prezentacji studenci przewidują pytania, jakie prawdopodobnie padną w toku dyskusji i opracowują odpowiedzi.

Sama prezentacja wiąże się z ogromną treścią, ale i olbrzymią mobilizacją. Wszyscy studenci podkreślają siłę emocji, jakich doświadczyli w trakcie prezentacji projektu – niezależnie od stopnia zaawansowania projektu, stopnia przygotowania zespołu i oczekiwań.

## 7. Najczęstsze problemy

Podstawowym problemem studentów informatyki jest "kodocentryczność". Za tym skolokwializowanym określeniem kryje się skłonność do zbyt technicznego spojrzenia na projekt i wynikające z niego problemy i niedostrzegania czynnika ludzkiego, nawyków, praktyk czy odczuć. Przejawem tego samego, stechniczowanego nastawienia, jest pisanie kodu zbyt wcześnie, zanim zespół dobrze zdiagnozuje problem lub zaprojektuje rozwiązanie.

Komunikowanie w grupie o własnych działaniach, zamiarach czy napotkanych problemach nie zawsze od początku działa. Rolą opiekuna jest wyjaśnić zespołowi w jaki sposób jawność i przekazywanie tego typu informacji w grupie przyczynia się do większej wydajności prac i postępu projektu. Typową sytuacją jest taka, kiedy jedna osoba utknie na trudnym do rozwiązania problemie i nie szuka pomocy u innych członków zespołu. Tymczasem zdarza się, że inna osoba zna już rozwiązanie danego problemu. Nawet jeśli nie, to krótka wymiana opinii może naprowadzić zespół na rozwiązanie.

Kolejnym problemem jest unikanie notatek, postrzeganych jako dodatkowy, niepotrzebny obowiązek. Przy rozpiętości czasowej projektu (6 miesięcy), ogromnym nakładzie prac i długiej liście problemów poleganie wyłącznie na pamięci prowadzi

do strat czasu. Wyrobienie nawyku notowania jest jednym z pierwszych problemów, przed jakim staje zespół.

Najpoważniejszym problemem jest jednak perspektywa czasowa. Z punktu widzenia studentów dwa semestry to mnóstwo czasu na projekt i skłonność do odkładania prac na później w początku zajęć jest w tej sytuacji dość naturalna. Odbija się ona jednak na całości projektu, ponieważ kompetencje członków zespołu rosną wykładniczo i pod koniec trwania przedmiotu są oni najbardziej efektywni. Wcześniejsze osiągnięcie wysokiego poziomu kompetencji zapewnia znacznie bardziej kompletne, staranniejsze wykonanie projektu i dopracowanie szczegółów.

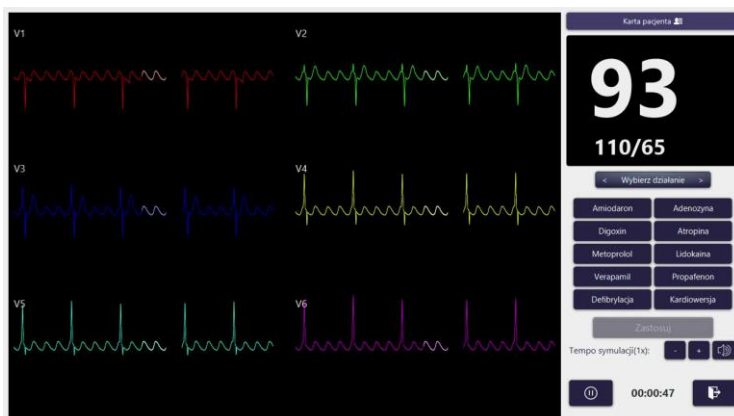
## 8. Moje doświadczenia z ostatnio prowadzonych projektów

W ostatnich dwóch latach byłem opiekunem dwóch zespołów, których projekty zostały nagrodzone w konkursie wydziałowym.

### 8.1. Sim Cor

Specyficznym projektem był prowadzony w roku akademickim 2015/2016 Sim Cor – symulator arytmii serca – w którym opiekowałem się zespołem wspólnie z dr Zofią Grąbczewską, lekarzem kardiologiem z Collegium Medicum UMK.

Zadaniem studentów było zrealizowanie symulatora przebiegu terapii schorzeń z rodziny arytmii serca z obrazem EKG oraz środkami terapeutycznymi w postaci podstawowych leków oraz kardiowersji i defibrylacji. Wymagało to stworzenia informatycznego modelu układu bodźcotwórczo-przewodzącego wraz z możliwymi odchyleniami od normy, modeli środków terapeutycznych z działaniem na układ bodźcotwórczo-przewodzący oraz wymodelowania krzywej EKG.



Rysunek 1. Ekran aplikacji Sim Cor

Podstawową trudnością projektu Sim Cor było odwzorowanie procesów zachodzących w wybranym układzie żywego organizmu. Zespół dokonał tego zdobywając odpowiednią wiedzę medyczną z literatury oraz w oku bezpośrednich, wielogodzinnych spotkań z konsultantem merytorycznym. W ich trakcie studenci zdobyli też wiedzę o procedurach nauczania studentów medycyny diagnostyki i leczenia arytmii serca oraz przyjętych praktykach w tym zakresie.

Drugą trudnością było znalezienie wspólnego języka między konsultantem medycznym a studentami informatyki. Na co dzień obie grupy posługują się diametralnie różnym aparatem pojęciowym, zupełnie w inny sposób opisują zjawiska. Dzięki staraniom obu stron komunikacja stała się płynna i skuteczna.

Zespół wykonał imponującą pracę, tworząc działający symulator arytmii serca dla środowiska komputerów stacjonarnych (desktop) i laptopów, pisząc około 20 000 linii kodu w języku Java i tworząc w tym języku ponad 180 klas. Studenci informatyki posiadli wiedzę na temat diagnozowania arytmii serca na podstawie EKG porównywalną z wymaganą od przyszłych lekarzy – już w trakcie trwania projektu rozpoznając na podstawie wykresu schorzenie. Studenci użyli bazy danych MySQL – kilku bibliotek i frameworków w Javie, pozostając przy narzędziach dobrze im znanych. Siłą projektu były stopień jego trudności technicznej oraz interdyscyplinarność.

Realizujący projekt zespół w składzie: Szymon Domeracki, Tomasz Grzona, Szymon Kumorek, Bernadeta Rybicka i Maciej Sikorski zdobył I miejsce w konkursie Programowania Zespołowego, w tym samym konkursie uzyskał również nagrody Prezydenta Miasta Torunia oraz Fundacji Rozwoju Informatyki. Na XXII Forum Teleinformatyki w sesji *Forum młodych mistrzów – Ekonomiczne aspekty informatyzacji państwa* we wrześniu 2016 roku zespół Sim Cor otrzymał Nagrodę Rady Programowej Forum Teleinformatyki w kategorii "Najlepsza merytorycznie prezentacja" oraz wyróżnienie Oddziału Mazowieckiego Polskiego Towarzystwa Informatycznego. Najlepsza merytorycznie prezentacja trwała równie 180 sekund i była najkrótszą w całej sesji.

O sukcesach zdecydowały wszechstronne zdolności zespołu, od programistycznych, poprzez znakomitą umiejętność zorganizowania pracy, skuteczne rozwiązywanie problemów, umiejętności prezentacyjne i zdolność do szybkiego zaadoptowania do warunków pracy.

## 8.2. BICOM

Celem projektu BICOM realizowanego w roku akademickim 2016/2017 było stworzenie aplikacji mobilnej na smartfony dla rowerzystów dostarczającej najbardziej potrzebnych informacji w połączeniu z najwygodniejszym, najbardziej ergonomicznym sterowaniem.



Zespół w składzie: Paulina Bawolska, Paweł Dembiński, Daniel Jaskólski, Mateusz Sałek, Jakub Sobański, Bartosz Wilczyński przygotował aplikację planując potrzebne funkcjonalności, sprawdzając poprawność planów poprzez ankietowanie potencjalnych użytkowników. Wśród najbardziej pożądaných funkcjonalności były: nawigacja, możliwość odbierania i wybierania połączeń bez trzymania telefonu w rękę, odtwarzanie muzyki, prognoza pogody (zespół „dołożył” do tego ostrzeżenia burzowe) oraz komunikowanie rowerzystów w grupie – z udostępnianiem położenia poszczególnych członków grupy rowerowej na mapie nawigacyjnej. Sterowanie aplikacją bez potrzeby trzymania telefonu w rękę zrealizowano głosowo, poprzez dołączany via Bluetooth kontroler oraz tradycyjnie – poprzez ekran dotykowy. Jako wiodących technologii zespół użył języka programowania Python i frameworku Kivy oraz dla przejęcia niektórych funkcji telefonu, takich jak odbieranie połączeń czy wysyłanie SMS-ów – języka Java.

W toku prac zespół wydał 4 wersje testowe aplikacji, które udostępniono użytkownikom. Zebrane od nich 38 uwag posłużyło dopracowaniu detali.

Główną zaletą projektu nie było zrobienie nowej aplikacji mobilnej (istnieje pokaźna liczba aplikacji rowerowych, szczególnie nawigacji i liczników), a integracja najbardziej potrzebnych funkcjonalności w przyjaznym interfejsie i wygodnie sterowanych.

Starania zespołu doceniło jury, przyznając mu II miejsce w konkursie.

## 9. Podsumowanie

Przez 18 lat Programowania Zespołowego zgromadziliśmy na Wydziale bogate doświadczenie w dziedzinie prowadzenia projektów studenckich i ich prezentacji. Programowanie Zespołowe daje studentom kierunków informatycznych unikalną okazję do zdobycia nowych umiejętności, przede wszystkim "miękkich", społecznych kompetencji w zakresie komunikacji i współpracy. Publiczne prezentacje odwiedzają przedstawiciele firm, szukając w gronie prezentujących studentów przyszłych pracowników.

Zespoły studenckie zdobywały kilkakrotnie nagrody na Forum Teleinformatyki. Znaczący czynnik w tych sukcesach miał fakt, że prezentowały gotowe, działające projekty i były już doświadczone w prezentowaniu projektu i w dyskusji z jury i publicznością.

Dla wielu naszych studentów przygoda z programowaniem zespołowym staje się fundamentem wiary we własne możliwości. Zdobycie doświadczenia (choć niektórzy funkcjonują już na rynku pracy) i możliwość wpisania do CV punktu o wygranym konkursie, jak i obecność przedstawicieli firm często przecierają drogę do przyszłej pracy.

## Literatura

1. Bawolska P., Dembiński P., Jaskólski D., Sałek M., Sobański J., Wilczyński B., *BICOM*, prezentacje Programowania Zespołowego, WMiI, UMK w Toruniu, 18.05.2017r.
2. Domeracki S., Grzona T., Kumorek S., Rybicka B., Sikorski M., *Sim Cor*, prezentacje Programowania Zespołowego, WMiI, UMK w Toruniu, 4.05.2016r., <http://tv.umk.pl/#movie=3086>
3. Domeracki S., Grzona T., Kumorek S., Rybicka B., Sikorski M., *Sim Cor – interaktywne szkolenie przyszłych lekarzy*, XXII Forum Teleinformatyki, Warszawa, 29-30.09.2016r.