Instytut Informatyki, Politechnika Łódzka

MGR INŻ. RAFAŁ WŁODARSKI,
PROMOTOR: DR HAB. INŻ. ANETA PONISZEWSKA - MARAŃDA, PROF. NADZW.

HYBRYDOWY CYKL ŻYCIA OPROGRAMOWANIA W PROJEKTACH AKADEMICKICH

PLAN PREZENTACJI

1. Wybór tematu oraz celu badań
2. Przeprowadzone i dalsze badania
3. Metody i narzędzia badawcze - schemat ewaluacji sukcesu przedsięwzięć programistycznych
4. Charakterystyka grup badawczych
5. Wyniki zrealizowanych badań
   - 5.1. Porównanie podejść iteracyjnego i sekwencyjnego
   - 5.2. Zaproponowanie podejścia hybrydowego
   - 5.3. Porównanie podejść iteracyjnego, sekwencyjnego i hybrydowego
6. Podsumowanie
7. Przyznane granty, opublikowane prace

TEMAT I CEL BADAŃ

- Definicja procesów produkcji oprogramowania - dwie główne rodziny: klasyczne i zwinne
- Zmieniające się środowisko komercyjne - ewolucja metodologii zarządzania projektami deweloperskimi
- Podejście hybrydowe
- Deficyt merytorycznych badań ewaluacji
- Ograniczona liczba badań podejścia hybrydowego do wykorzystania w środowisku akademickim

ZREALIZOWANE, AKTUALNE, DALSZE BADANIA

- Opracowanie metodologicznego badania sukcesu w przedsięwzięciach programistycznych, aby móc użyć go do porównania i oceny różnych procesów (rok akademicki 2016/2017)
- Adaptacja i wprowadzenie formalnego procesu produkcji oprogramowania w ramach zajęć praktycznych na Politechnice Łódzkiej (rok akademicki 2017/2018):
  - Zaadaptowany Scrum - *Zwinne metodyki prowadzenia projektów*
  - Zaadaptowany proces kaskadowy - *Modelowanie i analiza systemów informatycznych*
- Ewaluacja trzech typów procesów produkcji oprogramowania w ramach badań prowadzonych na dwóch uczelniach przy użyciu opracowanego schematu (rok akademicki 2018/2019)
  - Politechnika Łódzka (*Modelowanie i analiza systemów informatycznych),* Uniwersytet w Bordeaux *(Programmation Web)*
    - Zaadaptowany Scrum
    - Zaadaptowany proces kaskadowy
    - Grupa kontrolna (freestyle)
- Stworzenie hybrydowego procesu dla edukacyjnych projektów akademickich:
  - charakterystyka i organizacja faz procesu produkcji oprogramowania w trakcie semestru,
  - dostosowane praktyki na różnych etapach powstawania systemu.

ZREALIZOWANE, AKTUALNE, DALSZE BADANIA

- Porównanie opracowanego hybrydowego procesu z innymi podejściami w ramach zajęć *Programmation Web* na Uniwersytecie w Bordeaux (rok akademicki 2019/2020)
  - Proces hybrydowy
  - Proces iteracyjny
  - Proces sekwencyjny
- Powtórzenie powyższego eksperymentu w roku akademickim 2020/2021 aby potwierdzić hipotezy odnośnie opracowanego podejścia hybrydowego

METODY I NARZĘDZIA BADAWCZE

Określenie metodologicznego badania sukcesu przedsięwzięć programistycznych:

- przegląd literatury naukowej - kryteria oraz sposoby pomiarów ewaluujące rezultaty danego procesu tworzenia oprogramowania bądź inicjatywy jego poprawy

JAKOŚĆ W PROJEKCIE

- **wewnętrzna jakość produktu (kod źródłowy)**
- **zewnętrzna jakość produktu jak zdefiniowana przez standard ISO25010**

WYDAJNOŚĆ W PROJEKCIE

- **Funkcjonalna kompletność**
- **Funkcjonalna odpowiedniość**

CZYNNIKI SPOŁECZNE

- **jakość pracy zespołowej (spójność zespołu)**
- **rezultaty nauczania**

Użyte metody badawcze:

- **Ilościowe**: mierniki i metryki zbierane na podstawie kodu źródłowego, oraz końcowych projektów
- **Jakościowe**: obserwacja uczestnicząca, sondaże diagnostyczne, kwestionariusze, ankiety

METODY I NARZĘDZIA BADAWCZE

JAKOŚĆ W PROJEKCIE - WEWNĘTRZNA JAKOŚĆ PRODUKTU

KOD PHP

Cyclomatic Complexity

$$CC = \sum_{i=1}^{n} E_i - \sum_{j=1}^{nm} N_j + 1$$

Złożoność kodu, łatwość wykrywania defektów i pielęgnacji

+

- Lines of Code (LOC),
- duplicated LOC,
- Unit interfacing,
- Module coupling (zależność oprogramowania)

=

Maintainability rating

Software Improvement Group & TÜViT quality institute

ISO 25010

KOD HTML

HTML errors

E = A/B, E ∈ [0, 1]

gdzie A to liczba zwróconych błędów przez walidator W3C,
B – liczba linii kodu HTML

HTML warnings

W = A/B, E ∈ [0, 1]

gdzie A to liczba zwróconych ostrzeżeń przez walidator W3C,
B – liczba linii kodu HTML

JAKOŚĆ W PROJEKCIE - ZEWNĘTRZNA JAKOŚĆ PRODUKTU

- Obranie i adaptacja metryk mierzących wybrane właściwości oprogramowania zdefiniowanych przez standard **ISO 25010**

JAKOŚĆ PRODUKTU

PIELĘGNOWALNOŚĆ
PRZENASZALNOŚĆ
UŻYTECZNOŚĆ
FUNKCJONALNA ODPOWIEDNIOŚĆ
WYDAJNOŚĆ W DZIAŁANIU
NIEZAWODNOŚĆ
BEZPIECZEŃSTWO
ZGODNOŚĆ

JAKOŚĆ UŻYTKOWA

SATYSFAKCJA
WYDAJNOŚĆ
KONTEKST UŻYCIA
EFEKTYWNOŚĆ
WOLNOŚĆ OD RYZYKA

- Definicja wspólnego zestawu miar ewaluacji jakości dla wszystkich grup

APLIKACJE WEBOWE

EFEKTYWNOŚĆ W DZIAŁANIU

Czas

X = A

A = czas w milisekundach, który upłynął na wykonanie danego przypadku testowego

0 <= X < ∞
Im bliżej 0, tym lepiej

WYDAJNOŚĆ W DZIAŁANIU

Liczba kliknięć

X = A + B

A = liczba lewych, prawych i środkowych kliknięć myszkom*
B = liczba przewinięć myszki*

* w trakcie wykonywania danego przypadku testowego

0 <= X < ∞
Im bliżej 0, tym lepiej

WYDAJNOŚĆ W DZIAŁANIU

Przebyty dystans

X = A

A = pokonany dystans w milimetrach, w trakcie wykonywania danego przypadku testowego

0 <= X < ∞
Im bliżej 0, tym lepiej

WYDAJNOŚĆ W PROJEKCIE

FUNKCJONALNA ODPOWIEDNIOŚĆ

Funkcjonalna kompletność:

X = 1 - A/B, gdzie **A** = liczba brakujących funkcji,
B – liczba funkcji opisanych w specyfikacji
ZF ∈ **[0, 1],** im wyższy wynik tym lepiej

0 <= X <= 100%
Im bliżej 100%, tym rozwiązanie jest bardziej kompletne

FUNKCJONALNA ODPOWIEDNIOŚĆ

Funkcjonalna odpowiedniość

X = **1 - C/D**, gdzie C = liczba funkcji z anomaliami,
D – liczba funkcji opisanych w specyfikacji
X ∈ **[0, 1],** im wyższy wynik tym lepiej

0 <= X <= 100%
Im bliżej 100% tym lepiej

CZYNNIKI SPOŁECZNE

JAKOŚĆ PRACY ZESPOŁOWEJ

Komunikacja, koordynacja,
wzajemne wsparcie, spójność zespołu

Spójność zespołu (team cohesion)

The Group Environment Questionnaire

- Przywiązanie zespołu do projektu
- Przywiązanie wewnątrz zespołu
- Indywidualne przywiązanie do projektu
- Indywidualne przywiązanie do grupy

METODA BADAWCZA: ANKIETY

Periodyczna ocena na 4 stopniowej skali Likerta

1. Mój zespół jest zjednoczony w dążeniu do osiągnięcia sukcesu projektu
2. Członkowie zespołu przejawiają zgodne aspiracje co do przyszłości projektu
3. Uważam poziom pragnienia grupy do osiągnięcia sukcesu za zadowalający
4. Praca w moim zespole daje mi wystarczająco okazji do wykazania się
5. Prawdopodobnie pozostanę w kontakcie z częścią zespołu gdy zakończymy pracę nad projektem
6. Członkowie zespołu uczestniczą wspólnie w spotkaniach towarzyskich poza projektem
7. Integracja z członkami zespołu sprawia mi przyjemność
8. Mój zespół jest jedną z ważniejszych grup społecznych do których aktualnie należę

CZYNNIKI SPOŁECZNE

REZULTATY NAUCZANIA

Kompetencje techniczne:

- Projektowanie oprogramowania
- Modelowanie danych
- Programowanie

Umiejętności miękkie

- Komunikacja interpersonalna
- Zdolność do pracy w grupie

METODA BADAWCZA: ANKIETY – RUBRYKI

Ocena na 4 stopniowej skali Likerta

Umiejętność	Stopień realizacji			
1. Umiejętność organizacyjne i efektywnej pracy w grupie	**1.** Student rzadko stawia się na spotkania oraz nie wykazuje zaangażowania wobec projektu; wkłada minimalny nakład pracy aby osiągnąć wyznaczone zadanie; nie zawsze wywiązuje się ze zobowiązań. Student nie wykazuje chęci pomocy innym.	**2.** Student uczestniczy w większości spotkań grupy i realizuje swoje obowiązki i zadania z reguły bez opóźnień; niesie pomoc innym gdy jest o to poproszony.	**3.** Student zawsze uczestniczy w aktywnościach grupy i wywiązuje się na czas ze swoich zobowiązań projektowych i wobec innych; czasem dobrowolnie oferuje pomoc kolegom.	**4.** Student respektuje wszystkie zasady pracy zespołu, informuje innych o swoim progresie i pomaga w organizacji pracy projektowej/ spotkań zespołu. Jest proaktywny w poszukiwaniu rozwiązań dla innych członków zespołu bądź projektu.
2. Umiejętności interpersonalne i komunikacyjne	**1.** Student rzadko przejawia pozytywny stosunek do projektu/ swoich zadań/kolegów. Nie dzieli się swoimi własnymi pomysłami, zdarza mu się ignorować te wysuwane przez innych bądź wyrażać swój sprzeciw niedyplomatycznie.	**2.** Student wspomaga aktywności wspierające posuwanie się projektu i czasami wychodzi z własnymi inicjatywami; przeważnie jest neutralny w grupowym podejmowaniu decyzji.	**3.** Student czasem wprowadza w życie własne inicjatywy aby przyspieszyć realizację projektu. Student dzieli się swoim zdaniem w dyskusjach/trakcie podejmowania decyzji grupowych i potrafi przyznać się do błędu. Często przejawia pozytywny stosunek do projektu/ pracy zespołu.	**4.** Student często gra rolę mediatora przy grupowym podejmowaniu decyzji i ułatwia osiągniecie konsensusu. W zależności od sytuacji student rozluźnia atmosferę pracy, bądź motywuje innych do pracy.

przykładowa rubryka oceniająca umiejętności miękkie

CHARAKTERYSTYKA GRUP BADAWCZYCH 2018/2019

Politechnika Łódzka

Wydział Fizyki Technicznej, Informatyki i Matematyki Stosowanej

Studenci I roku studiów magisterskich

Modelowanie i Analiza Systemów Informatycznych

- **dobór kwotowy**: 46
  równomierne rozłożenie kwalifikacji wśród grup

Zmienne różnicujące:
- ukończony pierwszy tok studiów
- doświadczenie zawodowe
- kompetencje techniczne i znajomość narzędzi

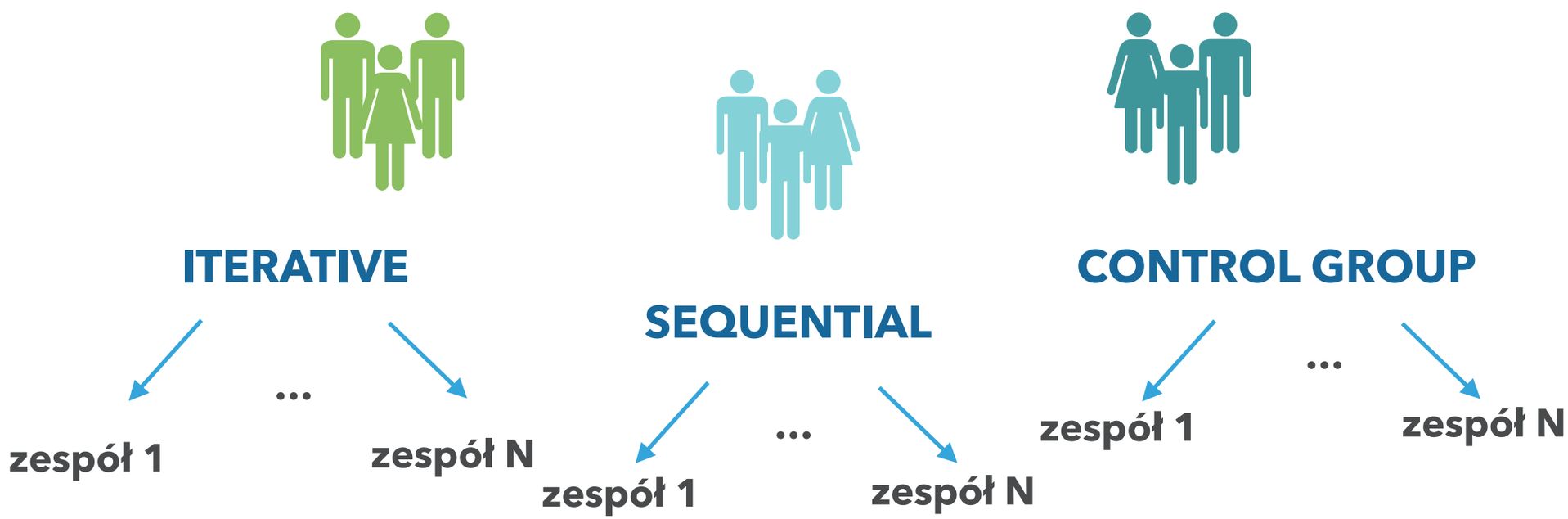
Uniwersytet w Bordeaux

Graduate School in Electronics, Computer Sciences, Telecommunications, Mathematics and Mechanics

Studenci III roku studiów inżynierskich

Programmation Web

- **losowy dobór próby**: 58
  - homogeniczne kompetencje techniczne studentów



ZREALIZOWANE BADANIA 2018/2019

ITERATIVE vs SEQUENTIAL

JAKOŚĆ W PROJEKCIE - WEWNĘTRZNA JAKOŚĆ PRODUKTU

Maintainability rating

0 <= X < 10
Im bliżej 10, tym lepiej

ITERATIVE	SEQUENTIAL	CONTROL
Case study 1 (FR)		
4.4	4.5	5.0
Case study 2 (PL)		
7.66	8.75	9.0

*Najlepszy wynik
*Najniższy wynik

HTML quality index

*X = (A + 0.5*B)*

(A) - number of errors*
(B) number of warnings*

* generated during a W3C validation for a pre- selected portion of pages

0 <= X < ∞
Im bliżej 0, tym lepiej

ITERATIVE	SEQUENTIAL	CONTROL
Case study 1 (FR)		
43.1	34.71	6.89

- Grupy kontrolne stworzyły rozwiązania o najwyższej jakości wewnętrznej w obydwu badaniach
- Grupy pracujące zgodnie ze Scrumem stworzyły rozwiązania o najniższej jakości wewnętrznej w obydwu badaniach

ZREALIZOWANE BADANIA 2018/2019

ITERATIVE vs SEQUENTIAL

JAKOŚĆ W PROJEKCIE - ZEWNĘTRZNA JAKOŚĆ PRODUKTU

EFEKTYWNOŚĆ W DZIAŁANIU

Czas:

0 <= X < ∞
Im bliżej 0, tym lepiej

	ITERATIVE	SEQUENTIAL	CONTROL
Case study 1 (FR)	17.84	**12.35**	17.73
Case study 2 (PL)	36.76	**25.78**	33.5

WYDAJNOŚĆ W DZIAŁANIU

Liczba kliknięć:

0 <= X < ∞
Im bliżej 0, tym lepiej

	ITERATIVE	SEQUENTIAL	CONTROL
Case study 1 (FR)	6.6	**3.7**	5.67
Case study (PL)	26.16	**12.47**	29.06

WYDAJNOŚĆ W DZIAŁANIU

Przebyty dystans:

0 <= X < ∞
Im bliżej 0, tym lepiej

	ITERATIVE	SEQUENTIAL	CONTROL
Case study (FR)	43.64	**29.25**	40.53
Case study (PL)	238.14	**153.25**	183.58

***Najlepszy wynik**
***Najniższy wynik**

- Grupy pracujące zgodnie z modelem kaskadowym stworzyły rozwiązania o najwyższej jakości zewnętrznej w obydwu badaniach

ZREALIZOWANE BADANIA 2018/2019

ITERATIVE vs SEQUENTIAL

WYDAJNOŚĆ W PROJEKCIE

FUNKCJONALNA ODPOWIEDNIOŚĆ

Funkcjonalna kompletność:

0 <= X <= 100%
Im bliżej 100%, tym rozwiązanie jest bardziej kompletne

ITERATIVE	SEQUENTIAL	CONTROL
Case study 1 (FR)		
60.0	46.43	42.86
Case study 2 (PL)		
100	100	85.71

Funkcjonalna odpowiedniość

0 <= X <= 100%
Im bliżej 100% tym lepiej

ITERATIVE	SEQUENTIAL	CONTROL
Case study 1 (FR)		
80.53	89.76	90.48
Case study 2 (PL)		
76.67	96.25	83.61

*Najlepszy wynik
*Najniższy wynik

- Grupy pracujące zgodnie ze Scrumem były najbardziej wydajne jednak miały najniższy wskaźnik funkcjonalnej odpowiedniości w obydwu badaniach

ZREALIZOWANE BADANIA 2018/2019

ITERATIVE vs SEQUENTIAL

JAKOŚĆ PRACY ZESPOŁOWEJ

Spójność zespołu (team cohesion)

*Najlepszy wynik
*Najniższy wynik

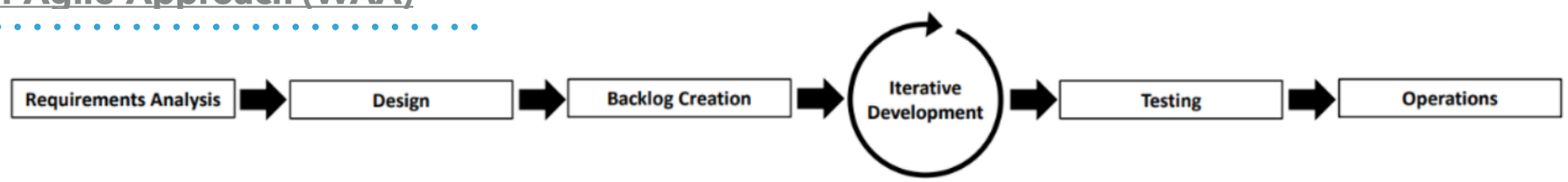
	ITERATIVE	SEQUENTIAL	CONTROL
	Case study 1 (FR)		
Positive perception	93.35	80.27	77.15
Negative perception	6.65	19.73	22.85
	Case study 2 (PL)		
Positive perception	78.84	67.64	69.68
Negative perception	21.16	32.36	30.32

- Zespoły pracujące zgodnie ze Scrumem wykazały się najwyższą spójnością zespołu (ok 10%) w obydwu badaniach

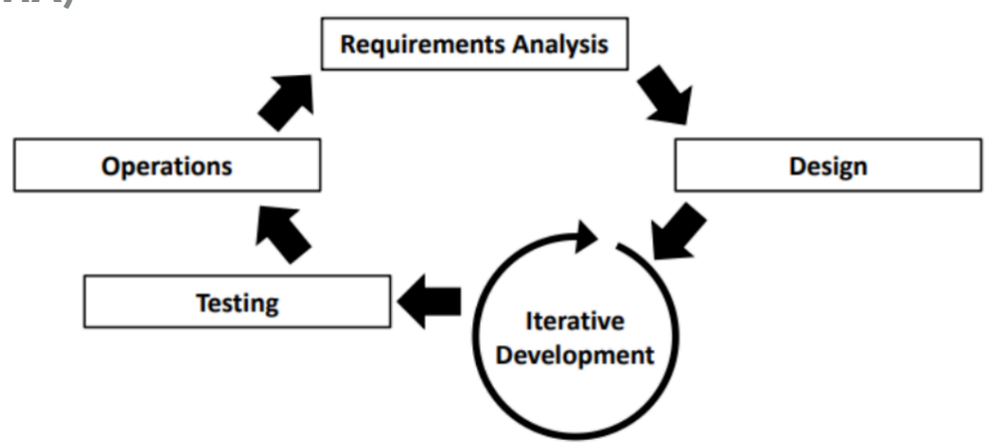
PODEJŚCIE HYBRYDOWE: KONTEKST KOMERCYJNY

- Zachowane wszystkie fazy cyklu życia wg. modelu Royce'a
- 3 sposoby organizacji metod hybrydowych*

Waterfall-Agile-Approach (WAA)



the Waterfall- Iterations-Approach (WIA)



the Pipeline-Approach (PA)

Requirements Analysis	N	N+1	N+2	N+3	N+4
Design	N-1	N	N+1	N+2	N+3
Iterative Development ↻	N-2	N-1	N	N+1	N+2
Testing	N-3	N-2	N-1	N	N+1
Operations	N-4	N-3	N-2	N-1	N
Iteration	**I-2**	**I-1**	**I**	**I+1**	**I+2**

- Kontekst akademicki: znajomość procesu, złożoność projektu, długość zajęć

*N. Prenner, C. Unger-Windeler, K. Schneider, How are Hybrid Development Approaches Organized? - A Systematic Literature Review, *Proceedings of the International Conference on Software and System Processes 2020*

PODEJŚCIE HYBRYDOWE: KONTEKST AKADEMICKI

- **Introduce a dedicated design phase:** foundation of software's design is put in place and documented before the implementation phase kicks off.

- **Introduce iterative development:** implement a system through repeated cycles and in smaller portions at a time in order to retrieve feedback and enable continuous learning.

- **Introduce a dedicated testing phase**: formalised testing activities ensure high quality of the system delivered and address an important knowledge gap of software engineers.

Phase	Class	Activities
Design	1.	Course & way of working introduction; design initiation
Iteration I	2.	Design review, iteration I planning
Iteration II	3.	Demo, iteration II planning
Iteration III	4.	Demo, Iteration III planning
Test	5.	Testing session
Maintenance	+2W	Final hand in

PODEJŚCIE HYBRYDOWE: KONTEKST AKADEMICKI

PRAKTYKI

- **Start every iteration by a planning exercise:** the purpose of planning is to facilitate communication and control by defining the means of delivering the project.

- **Risk-based milestones**: this means choosing to develop features that represent the biggest business, programmatic or technical risks (and delivering the most value) while prioritising the work for each iteration.

- **Visualize and limit WIP (Work In Progress) on number of ongoing tasks**: by setting a limit and monitoring a maximum number of implementation tasks, the team keeps a sustainable pace throughout the semester and avoids nightly hackathons preceding milestones delivery.

- **Keep it simple tooling-wise**: each new technology or tool introduces overhead for students as they need to learn how to properly use it, thus drawing away their attention fro; the actual assignment and potentially hindering effective collaboration.

- **Team leader role**: without a person that provides direction and guidance to a group of developers, chances of achieving course's goal are slim. A designated team leader coordinates work of his team, helps resolve conflicts, tracks WIP and oversees tasks management.

- **Align team structure with architecture**: team roles correspond to modules that need to be developed. Individual module ownership means that distinct pieces or grouping of code are assigned to a single owner. The owner is responsible for the consistency, performance, and conceptual integrity of the module.

PODEJŚCIE HYBRYDOWE: KONTEKST AKADEMICKI

OGRANICZENIA

- **Organizacja kursu**
  - **Długość:** między 5 a 15 tygodniami aby móc wprowadzić minimum trzy fazy projektu i dwie iteracje
  - **Nakład pracy:** minimum 2 punkty ECTS
- **Skład zespołu**:
  - **Rozmiar:** między 3 a 10 osób; przy dużych zespołach należy zwiększyć stopień formalizacji na poziomie dokumentacji i komunikacji między członkami
  - **Umiejętności**: brak twardych warunków wstępnych aby móc zastosować proces; minimalne umiejętności korzystania z repozytorium tak jak w każdym programistycznym projekcie grupowym

HIPOTEZY

JAKOŚĆ PRODUKTU

- **HIPOTEZA**: Wprowadzenie formalnej fazy projektowania podniesie jakość zewnętrzną powstałego oprogramowania
  - **Efektywność w działaniu**
  - **Wydajność w działaniu**
- **HIPOTEZA**: Praktyka "Align team structure with architecture" pozytywnie wpłynie na wewnętrzną jakość
  - **Maintainability rating**
  - **HTML error index**

WYDAJNOŚĆ W PROJEKCIE

- **HIPOTEZA**: Wprowadzenie iteracji implementacyjnych podniesie efektywność zespołu
  - **Funkcjonalna kompletność**
- **HIPOTEZA**: Wprowadzenie formalnej fazy testowania podniesie niezawodność powstałego oprogramowania
  - **Funkcjonalna odpowiedniość**

CZYNNIKI SPOŁECZNE

- **HIPOTEZA**: Praktyka "Align team structure with architecture" będzie sprzyjała nabyciu kompetencji technicznych
  - **Rezultaty nauczania**

CHARAKTERYSTYKA GRUP BADAWCZYCH 2019/2020

Uniwersytet w Bordeaux

Graduate School in Electronics, Computer Sciences, Telecommunications, Mathematics and Mechanics

Studenci III roku studiów inżynierskich

Programmation Web

- **losowy dobór próby**: 66
  - homogeniczne kompetencje techniczne studentów

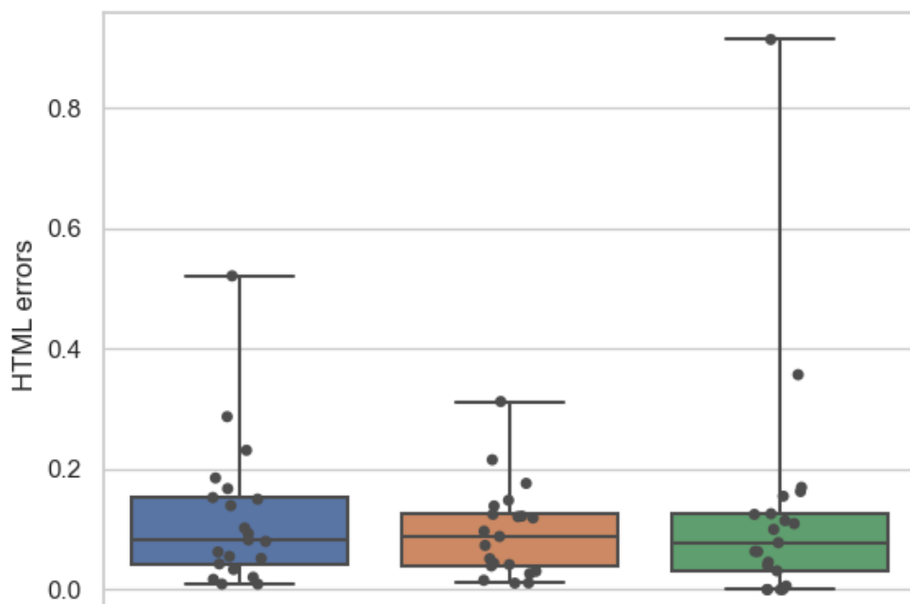
- **Podejście ITERACYJNE (GA): 7 zespołów 3 osobowych**
- **Podejście SEKWENCYJNE (GB): 7 zespołów 3 osobowych**
- **Podejście HYBRYDOWE (GC): 7 zespołów 3 osobowych**

ITERATIVE vs SEQUENTIAL vs HYBRID

JAKOŚĆ W PROJEKCIE - WEWNĘTRZNA JAKOŚĆ PRODUKTU

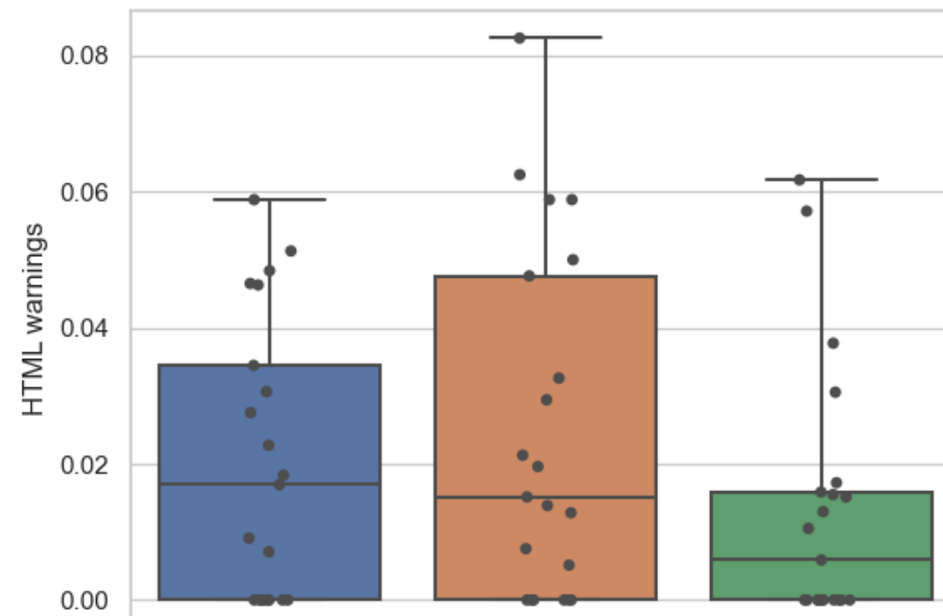
HTML Errors

0 <= X <1
Im bliżej 0, tym lepiej



HTML Warnings

0 <= X <1
Im bliżej 0, tym lepiej



- Zespoły hybrydowe stworzyły rozwiązania o najwyższej jakości wewnętrznej pod względem mediany ilości błędów i ostrzeżeń; niewielkie różnice (<10%)

Maintainability rating

0 <= X < 10
Im bliżej 10, tym lepiej

ITERATIVE	SEQUENTIAL	HYBRID
6.2	6	5,1

- Zespoły iteracyjne wypadły najlepiej pod względem jakości kodu PHP; zbliżony wynik dla zespołów sekwencyjnych
- Podejście hybrydowe wypadło najsłabiej, jednak wyniki nie zgadzają się z auto-ewaluacją studentów

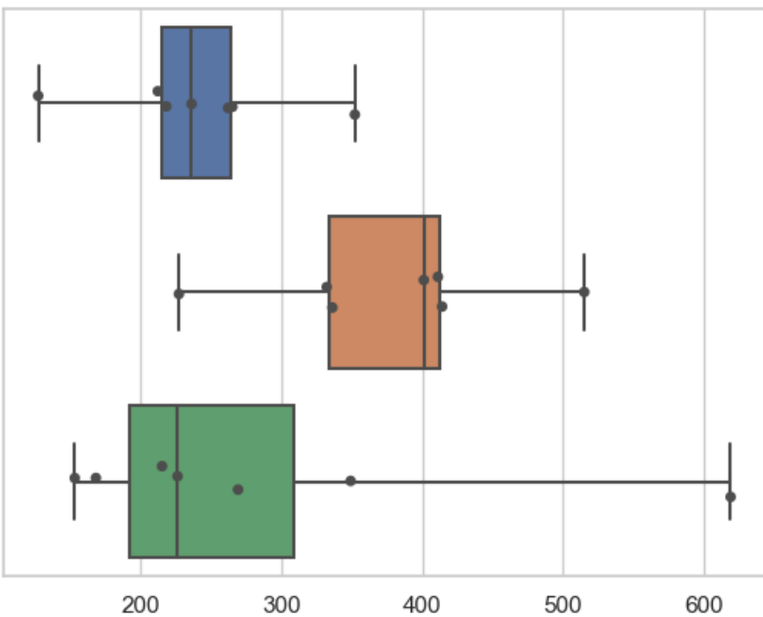
ZREALIZOWANE BADANIA 2019/2020

ITERATIVE vs SEQUENTIAL vs HYBRID

JAKOŚĆ W PROJEKCIE - ZEWNĘTRZNA JAKOŚĆ PRODUKTU

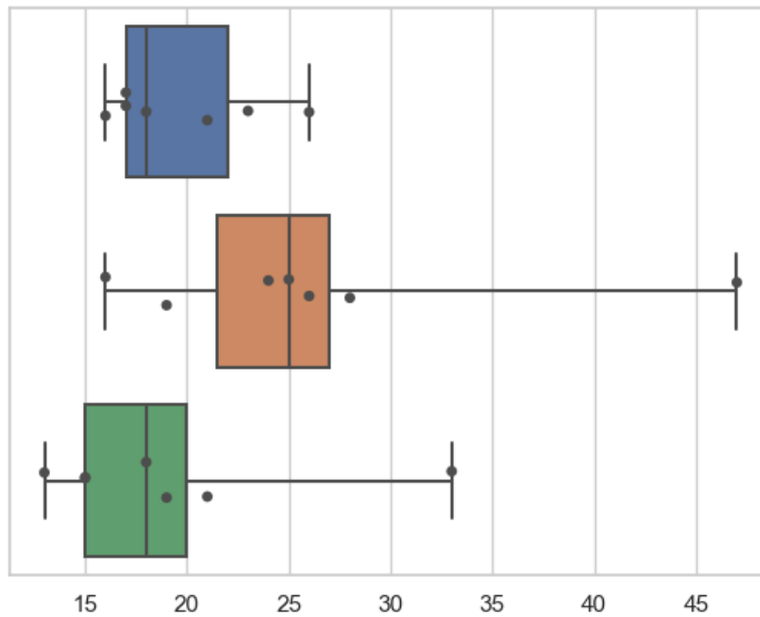
EFEKTYWNOŚĆ W DZIAŁANIU

Czas:



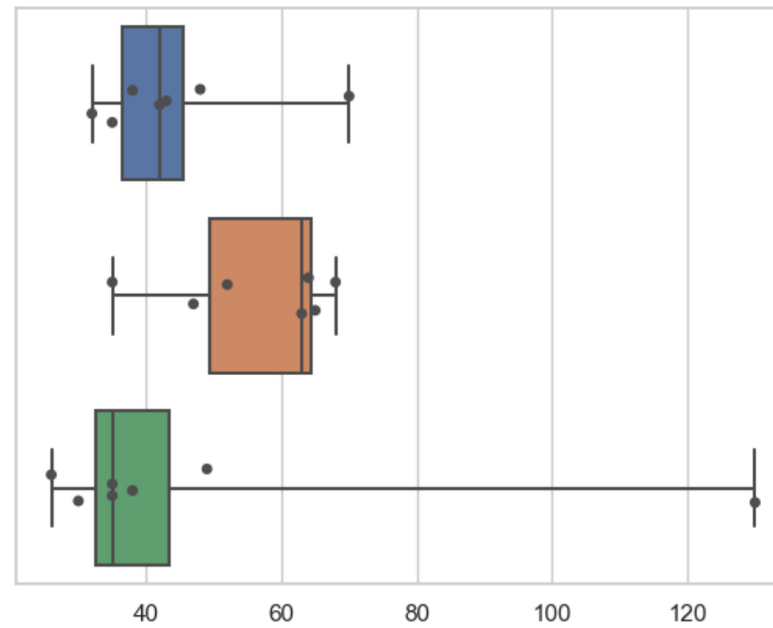
WYDAJNOŚĆ W DZIAŁANIU

Liczba kliknięć:



WYDAJNOŚĆ W DZIAŁANIU

Przebyty dystans:



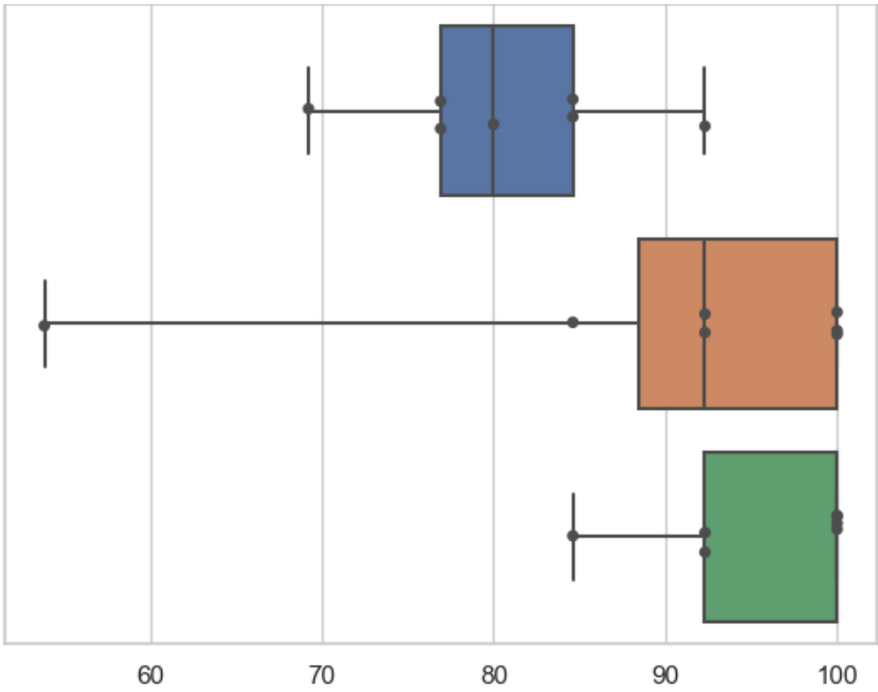
0 <= X <∞ Im bliżej 0, tym lepiej

- Brak spójnych rezultatów - iteracyjne i hybrydowe zespoły wypadły podobnie, sekwencyjne najsłabiej
- Wysunięta hipoteza nie może być potwierdzona jednak:
  - Grupa sekwencyjna była nadzorowana przez innego instruktora - udzielona różna informacja zwrotna odnośnie designu
  - Negatywny wpływ na ewaluowane metryki (brak znajomości aplikacji = dłuższy czas nauki)

ITERATIVE vs SEQUENTIAL vs HYBRID

WYDAJNOŚĆ W PROJEKCIE

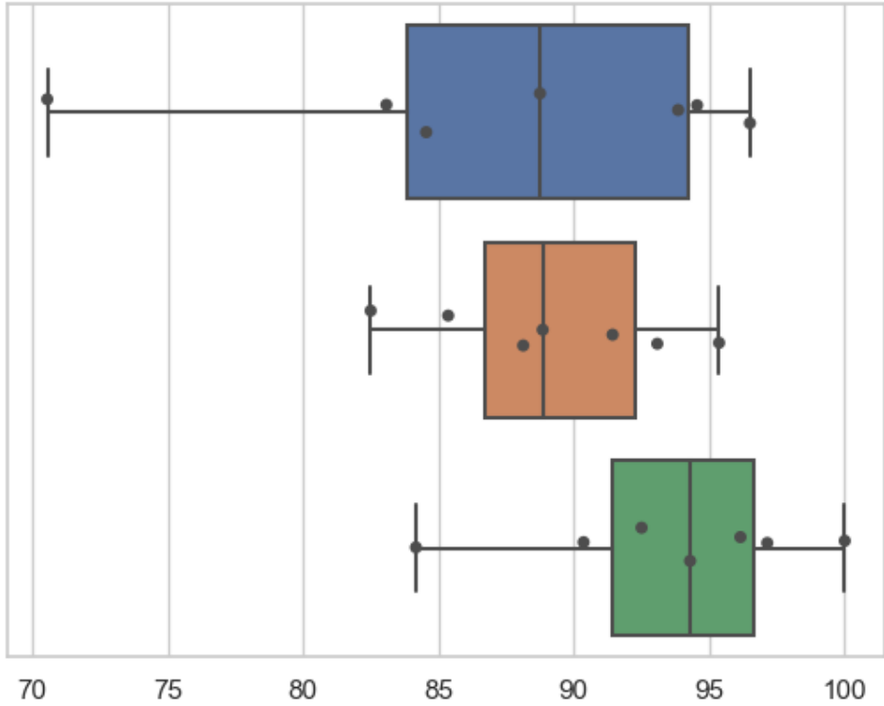
Funkcjonalna kompletność:



FUNKCJONALNA ODPOWIEDNIOŚĆ

0 <= X <= 100%;
im bliżej 100% tym lepiej

Funkcjonalna odpowiedniość

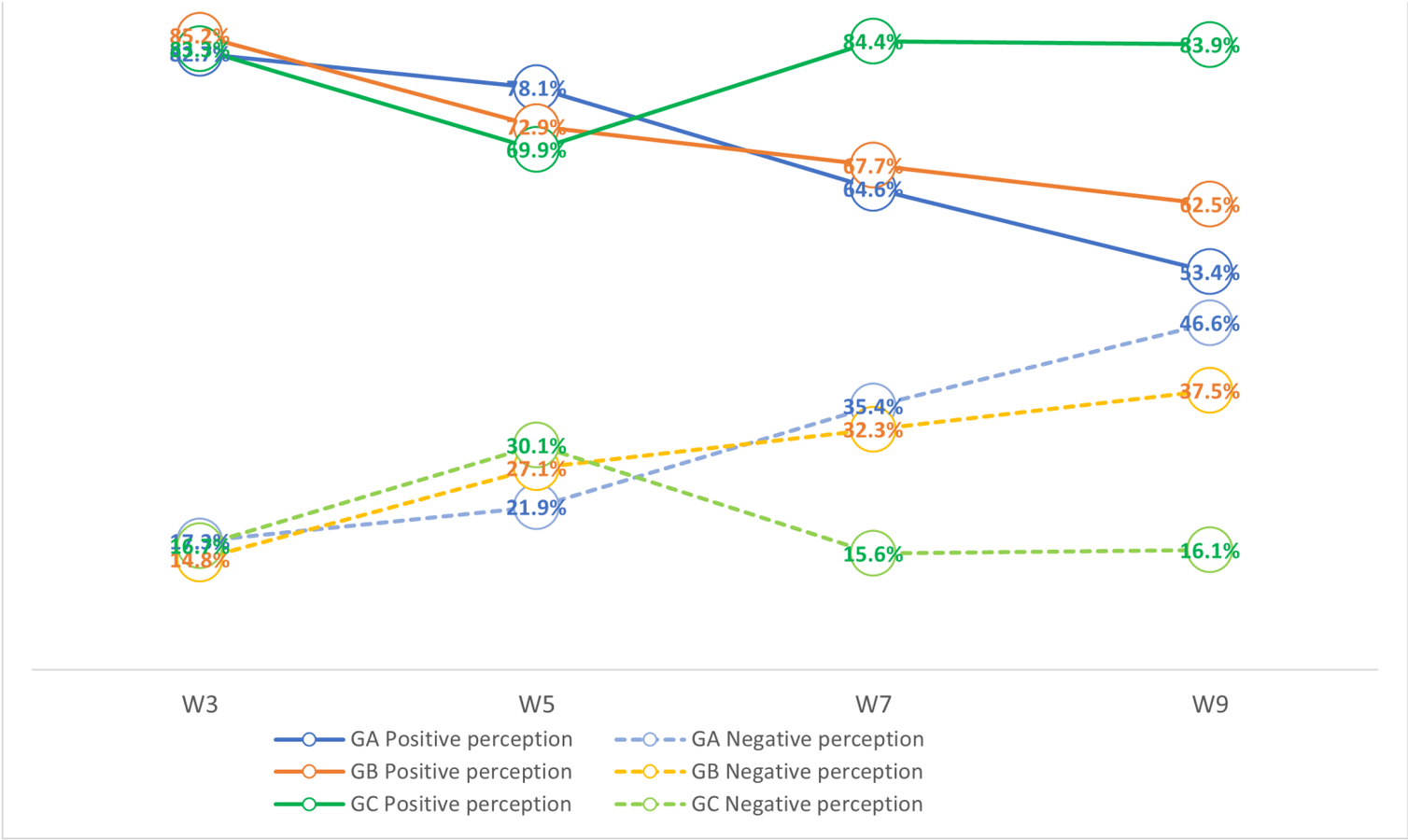


- Mediana FK dla podejścia iteracyjnego znacznie niższa (80%) niż dla sekwencyjnego (92%) i hybrydowego (100%) - wynik niespójny z poprzednim badaniem
- Najwyższy rozstęp miedzykwartylowy FK i FO dla zespołów hybrydowych (najlepszy wynik)
- Rozkład normalny FK dla grupy iteracyjnej sugeruje, że to podejście zapewnie stosunkowo regularny postęp prac
- Zbliżone wyniki FO dla grup iteracyjnych i sekwencyjnych (mediana 88.8 vs 88.9, średnia 87.4 vs 89.0)
  - Wprowadzenie fazy testowania nie jest wystarczające aby zapewnić sukces w tym wymiarze
  - Połączone z iteracyjną implementacją zapewnia systematyczne przetestowanie większej ilości funkcji

ZREALIZOWANE BADANIA 2019/2020

ITERATIVE vs SEQUENTIAL vs HYBRID

JAKOŚĆ PRACY ZESPOŁOWEJ Spójność zespołu (team cohesion)

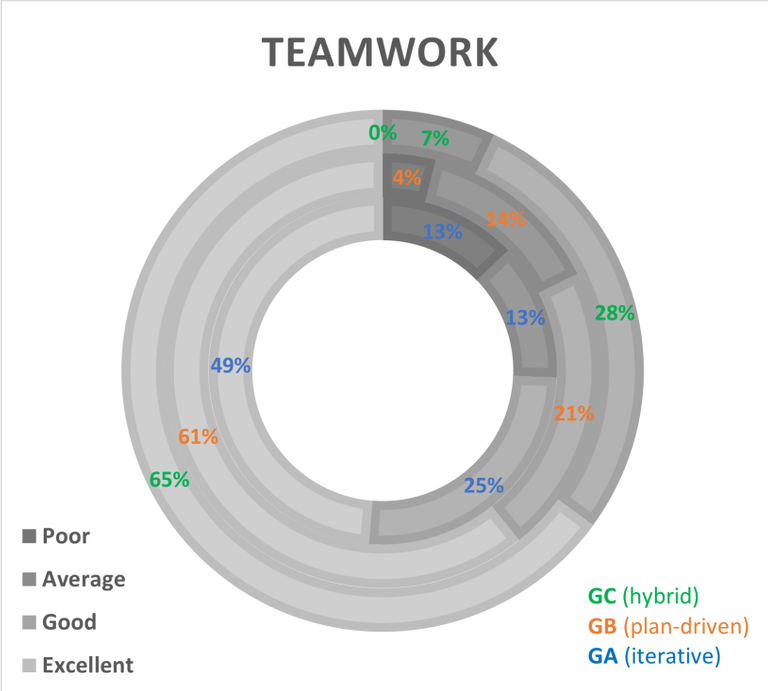


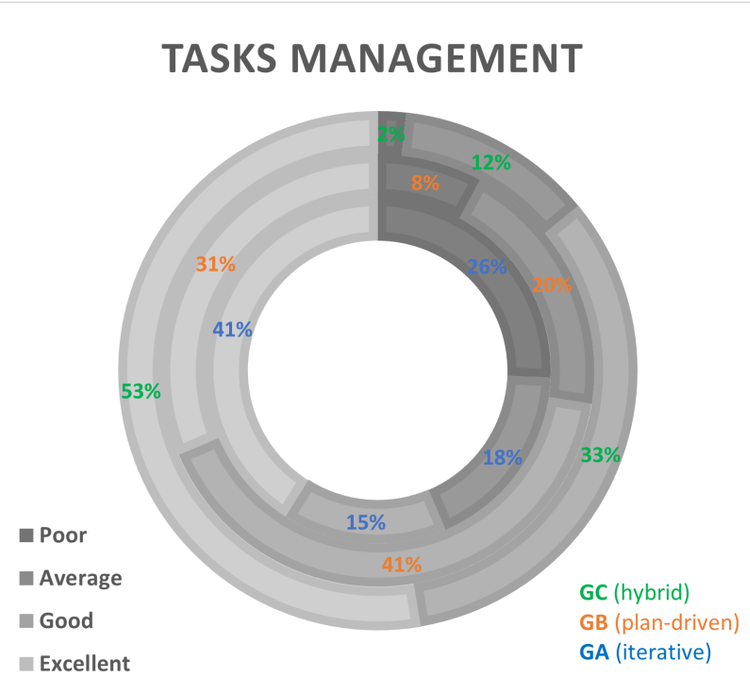
- Grupa hybrydowa jako jedyna nie wykazała tendencji spadkowej
  - Spadek w 5tym tygodniu prawdopodobnie odpowiadał trudnościom związanym z przejściem z designu do implementacji - grupa sekwencyjna wykazała podobną tendencję
- Grupa iteracyjna wykazała najniższą spójnością zespołu - wynik przeciwny do wcześniejszych badań
  - Dwa zespoły zgłosiły problem pracy w grupie (techniczne i zdrowotne) co mogło wpłynąć na wyniki; jednakże z punktu widzenia statystycznego, każda grupa mogła mieć podobne kłopoty

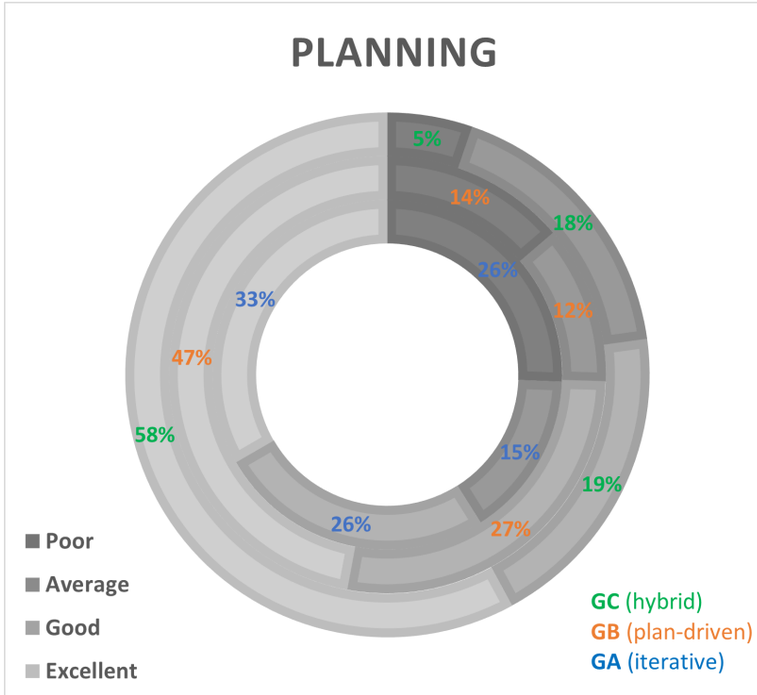
ZREALIZOWANE BADANIA 2019/2020

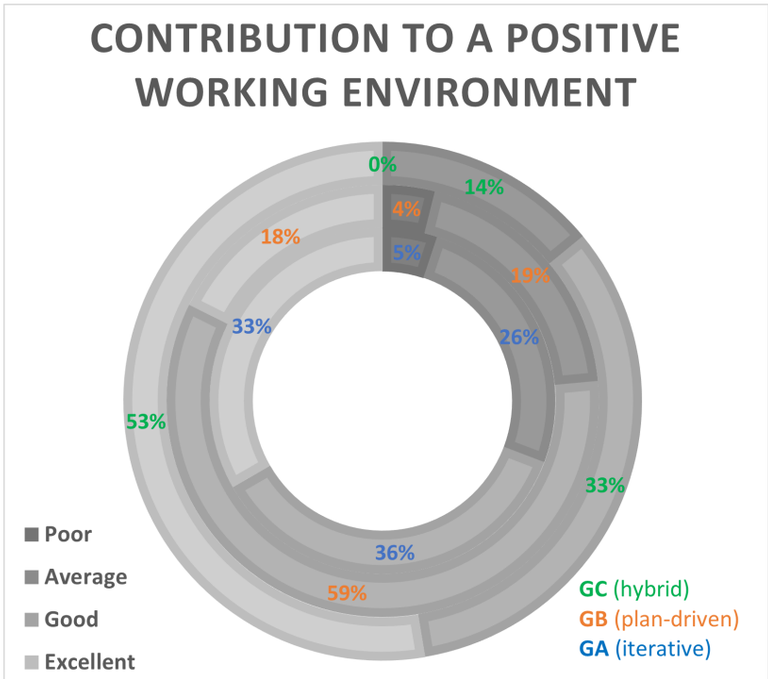
ITERATIVE vs SEQUENTIAL vs HYBRID

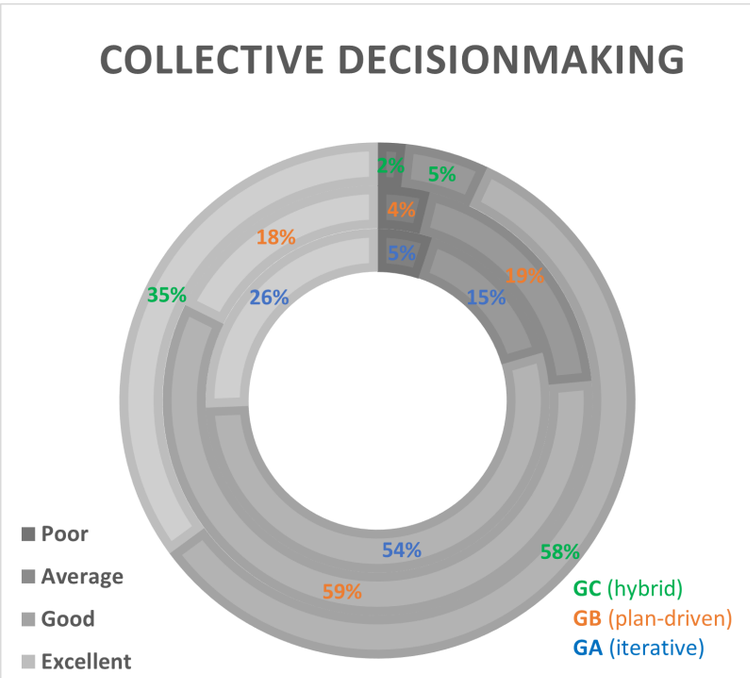
REZULTATY NAUCZANIA Umiejętności miękkie











OVERALL	GA	GB	GC
POSITIVE	65,81%	73,86%	85,67%
NEGATIVE	34,19%	26,14%	14,33%

- Wyniki bazujące na własnej ewaluacji oraz kolegów z zespołu
- Grupa hybrydowa wykazała najwyższe wyniki w każdej kategorii
- Problemy dwóch zespołów iteracyjnych mogły negatywnie wpłynąć na wyniki całej grupy

ZREALIZOWANE BADANIA 2019/2020

ITERATIVE vs SEQUENTIAL vs HYBRID

REZULTATY NAUCZANIA Umiejętności techniczne

HTML	GA	GB	GC
POSITIVE	100,00%	100,00%	94,74%
NEGATIVE	0,00%	0,00%	5,26%

CSS	GA	GB	GC
POSITIVE	61,54%	82,35%	73,68%
NEGATIVE	34,46%	17,65%	26,32%

PHP	GA	GB	GC
POSITIVE	84,62%	94,12%	94,74%
NEGATIVE	15,38%	5,88%	5,26%

BDD	GA	GB	GC
POSITIVE	84,62%	94,12%	89,47%
NEGATIVE	15,38%	5,88%	10,53%

SQL	GA	GB	GC
POSITIVE	100,00%	94,12%	94,74%
NEGATIVE	0,00%	5,88%	5,26%

*Najlepszy wynik
*Najniższy wynik

OVERALL	GA	GB	GC
POSITIVE	86,15%	92,94%	89,47%
NEGATIVE	13,85%	7,06%	10,53%

- Wyniki bazujące na auto-ewaluacji
- Brak spójnych wyników pomiędzy poszczególnymi kompetencjami a stosowanym podejściem
- Zbiorcze wyniki zbliżone pomiędzy wszystkimi grupami (różnica <6% pomiędzy najlepszym i najgorszym wynikiem)
- Ustanowienie związku między nabytymi kompetencjami technicznymi a stosowanym procesem jest dyskusyjne

PODSUMOWANIE

- Niektóre z wysuniętych hipotez zostały częściowo potwierdzone:
  - "Wprowadzenie formalnej fazy testowania podniesie niezawodność powstałego oprogramowania" **-> połączenie z iteratywną implementacją daje wyższe wyniki**
  - "Wprowadzenie iteracji implementacyjnych podniesie efektywność zespołu" **-> jedynie gdy połączone z formalną fazą testowania**
  - "Praktyka "Align team structure with architecture" pozytywnie wpłynie na wewnętrzną jakość oprogramowania" **-> niewielkie różnice w wynikach a także brak silnych dowodów na przestrzeganie praktyki**
- Część hipotez nie została potwierdzona:
  - "Wprowadzenie formalnej fazy projektowania podniesie jakość zewnętrzną powstałego oprogramowania" **-> wyniki zakłócone przez konfigurację eksperymentu**
  - Praktyka "Align team structure with architecture" będzie sprzyjała nabyciu kompetencji technicznych **-> trudno ustalić związek między stosowanym podejściem a rezultatami nauczania**
- Przeprowadzenie kolejnego eksperymentu potencjalnie pomoże potwierdzić powyższe hipotezy
- Dodatkowe badania potrzebne aby:
  - Móc generalizować wyniki (inne domeny rozwijanych aplikacji)
  - Móc ustalić związek między stosowanymi praktykami i ich wpływem na sukces projektu - poza zakresem pracy doktorskiej

WYBRANE PUBLIKACJE, GRANTY

1. *R. Włodarski, A. Poniszewska-Maranda, JR. Falleri, Comparative Case Study of Plan-Driven and Agile Approaches in Student Computing Projects*
Proceedings of SoftCOM 2020: International Conference on Software, Telecommunications and Computer Networks, September 2020, Split, Croatia

2. *R. Włodarski, A. Poniszewska-Maranda, Applying a Traditional Software Development Process to Drive Projects in Higher Education*
Proceedings of 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), September 2019, Thessaloniki, Greece

3. *R. Włodarski, A. Poniszewska-Maranda,* Measuring dimensions of Software Engineering projects' success in Academic context
Proceedings of FedCSIS 2017, Federated Conference On Computer Science And Information Systems, September 2017, Prague, Czech Republic

4. Projekt badawczy - Uniwersytet w Bordeaux (rok akademicki 2018/2019)

dofinansowany przez Projekt PROM – Międzynarodowa wymiana stypendialna doktorantów i kadry akademickiej

MGR INŻ. RAFAŁ WŁODARSKI,
PROMOTOR: DR HAB. INŻ. ANETA PONISZEWSKA - MARAŃDA, PROF. NADZW.

DZIĘKUJĘ ZA UWAGĘ