



Politechnika Wrocławska

Wydział Informatyki i Zarządzania

Kierunek studiów: Informatyka

Praca dyplomowa – inżynierska

**APLIKACJA WSPOMAGAJĄCA ZDALNE SZACOWANIE
HISTORYJEK UŻYTKOWNIKA METODĄ
PLANISTYCZNEGO POKERA.**

Karol Kowalski

słowa kluczowe:
Firebase, ReactJS, FLUX, planning poker, es-
tymacja

krótkie streszczenie:

W pracy przedstawiono projekt aplikacji umożliwiającej przeprowadzenie rozgrywki planning pokera online z użyciem ReactJS oraz Firebase.

Opiekun pracy dyplomowej	dr. hab. inż. prof. PWr. Trawiński Bogdan
	Tytuł/stopień naukowy/imię i nazwisko	ocena	podpis
Ostateczna ocena za pracę dyplomową			
Przewodniczący Komisji egzaminu dyplomowego	
	Tytuł/stopień naukowy/imię i nazwisko	ocena	podpis

*Do celów archiwalnych pracę dyplomową zakwalifikowano do:**

- a) kategorii A (akta wieczyste)*
- b) kategorii BE 50 (po 50 latach podlegające ekspertyzie)*

** niepotrzebne skreślić*

pieczętka wydziałowa

Wrocław, rok 2018

SPIS TREŚCI

Spis rysunków	2
Spis listingów	3
Spis tabel	4
Streszczenie	5
Wstęp	6
Opis problemu	6
Cel pracy	6
Zakres pracy	6
1. Wprowadzenie	7
1.1. Najpopularniejsze metody zarządzania projektami	7
1.2. Podejście klasyczne	8
1.3. Podejście zwinne	8
1.4. Manifest Agile	9
2. Najczęściej wykorzystywane narzędzia w zarządzaniu projektami.	12
2.1. Narzędzia stosowane dla metod klasycznych.	12
2.1.1. Microsoft Project	12
2.1.2. Gantt Project	12
2.2. Najpopularniejsze narzędzia stosowane dla metod zwinnych (team estimation game, kanban, planning poker).	13
2.3. Odniesienie do książki	14
2.3.1. Rysunek z kotem	14
2.3.2. Tabela	15
2.4. Listing	16
Zakończenie	18
Dodatki	19
A. To powinien by? dodatek	20
Bibliografia	21

SPIS RYSUNKÓW

1.1.	Manifesto for Agile Software Dev.[www.medium.com]	10
2.1.	Microsoft Project[www.microsoft.com]	13
2.2.	Ala ma kota (opr.wł).	15

SPIS LISTINGÓW

2.1	Przykładowy algorytm w języku C (opr. wł.)	17
-----	--	----

SPIS TABEL

1.1. Różnice między podejściem klasycznym a zwinnym	8
2.1. Co kto ma [?] (patrz też dodatek A)	16

STRESZCZENIE

Celem pracy było opracowanie aplikacji zaprojektowanej do gry w planning pokera zdalnie. Dostępne aplikacje spełniają swoje zadanie, jednak autor nie znalazł aplikacji, która byłaby silnie zintegrowana z Github'em. Dlatego wyróżniającą cechą jego aplikacji będzie możliwość importu historyjek z Github'a które są tam w postaci issues oraz exportu ocen do Github'a w postaci etykiet. W ramach pracy autor stworzył aplikację opartą o architekturę FLUX z bazą danych firebase jako backend. Dzięki temu rozwiązaniu planning poker może być rozgrywany w czasie rzeczywistym. Oprócz tego praca zawiera omówienie pewnych metod zarządzania zwinnego. Użyteczność aplikacji będzie sprawdzona w oparciu o wyniki ankiety, które zostaną zawarte w niniejszej pracy.

ABSTRACT

The main goal of this thesis was development of web app designed to play planing poker online. Avaliable applications fulfil their role but however author haven't found application, which would be strongly integrated with Github. That's why distinctive feature of author's app will be possibility is importing user stories from Github projects which are there in the shape of issues and exporting story points to Github in form of labels. Within work author made application based on FLUX architecture with firebase database as backend. Thanks this solution planning poker can be played in real time. Except of app description thesis describe some of agile managment and estimating methods. Usability of application will be tested thanks usability survey which reasults are consist within thesis.

WSTĘP

Planowanie to odpowiadanie na pytanie "Co powinniśmy stworzyć i kiedy?". Jednak aby odpowiedzieć na to pytanie powinniśmy zadać również pytania o estymację ("Jak duże to jest?") oraz harmonogram ("Kiedy będzie skończone?" oraz "Ile będzie zrobione do tego czasu?"). Estymowanie i planowanie są bardzo istotne w sukcesie każdego projektu. Plany pomagają inwestorom podjąć decyzję. Na przykład możemy zacząć specyficzny projekt, jeżeli oszacujemy, że zajmie sześć miesięcy oraz będzie wymagać milion dolarów, ale odrzucimy go, jeżeli stwierdzimy, że zabierze nam dwa lata oraz 4 miliony dolarów.[3]

OPIS PROBLEMU

W dzisiejszym świecie coraz więcej projektów jest tworzonych przez zespoły rozproszone. Zespół rozproszony to taki, którego członkowie realizują jeden projekt, cel i zadania w określonym czasie, pracując z różnych miejsc – kontynentów, krajów, miast, budynków czy biur. Zespoły rozproszone pracują jak tradycyjne zespoły zadaniowe, ale mają ze sobą na co dzień kontakt wirtualny i widują się sporadycznie.[13] Przez co nie mogą się spotykać co sprint w jednym pomieszczeniu by oszacować zadania do wykonania w sprincie. Dlatego muszą się spotkać w wirtualnym pokoju w chmurze. Jednak muszą też swoje estymacje mieć zapisane w centralnym miejscu, gdzie mieści się ich projekt. Bardzo często tym miejscem jest np. Github.

CEL PRACY

Zadaniem jakie jakie postawiłem przed sobą jest stworzenie aplikacji umożliwiającej rozegranie planning pokera w czasie rzeczywistym w raz z rolami product ownera, scrum mastera oraz gracza by jak najwierniej zasymulować rozgrywkę w planning pokera. Dodatkowym celem jest synchronizowanie tego wszystkiego z aplikacją Github.

ZAKRES PRACY

Praca obejmowała opracowanie projektu aplikacji, implementację w frameworku ReactJS oraz wdrożenie wszystkiego powyższego w bazie danych Firebase oraz zhostowanie tego wszystkiego na hostingu Firebase'a. Dodatkowym zadaniem jest sprawdzenie użyteczności programu za pomocą ankiety.

1. WPROWADZENIE

Najpopularniejszą definicją projektu jest definicja Project Management Institute (PMI – międzynarodowe stowarzyszenie zrzeszające kierowników projektów. Project Management Institute powstał w 1969 w Pensylwanii w USA jako stowarzyszenie non profit zrzeszające profesjonalistów w dziedzinie zarządzania projektami- istnieje również oddział we Wrocławiu.): „Projekt, to tymczasowa działalność podejmowana w celu wytworzenia unikatowego wyrobu, dostarczenia unikatowej usługi lub otrzymania unikatowego rezultatu”. [5] Już w starożytnym Egipcie istniały metody zarządzania skomplikowanym przedsięwzięciem, np. budowa piramid. Było to olbrzymie wyzwanie, które wymagało wiedzy zarówno planistycznej, jak i logistycznej. W ubiegłym stuleciu, w latach 50 stosowano podejścia zwane obecnie współczesnymi technikami zarządzania projektami. Weźmy np. projekt systemu rakiet balistycznych Polaris. Okazał się on swoistym koszmarem technicznym i administracyjnym. Nad projektem pracowała olbrzymia ilość zespołów badawczych, projektowych i produkcyjnych. Dla udokumentowania wszystkich działań zużyto tony papieru, a samo zarządzanie projektami zaczęto uznawać za dziedzinę bardzo skomplikowaną, niedostępną, opartą na wiedzy specjalistów. [12]

1.1. NAJPOPULARNIEJSZE METODY ZARZĄDZANIA PROJEKTAMI

Przy dokonywaniu wyboru metodyki zarządzania projektem należy przeprowadzić adekwatną analizę w celu doboru odpowiedniego podejścia, gdyż każda z metodyk posiada wady i zalety. Wyróżniamy dwa podejścia (klasyczne i zwinne), które różnią się dość mocno między sobą w kilku płaszczyznach (przekrojach), takich jak: odpowiedzialność za produkt, rola menedżera w zespole, istota prac wstępnych, zdefiniowanie produktu czy odpowiedź zwrotna użytkowników. Co uwzględniono w tabeli 1.1.

Tabela 1.1. Różnice między podejściem klasycznym a zwinnym

Płaszczyzna	Podejście klasyczne	Podejście zwinne
Odpowiedzialność za produkt	Podzielona między marketera, menadżera produktu i menadżera projektu	Istnieje tylko jeden właściciel produktu
Rola menedżera w zespole	oddzielony od zespołów deweloperskich	Jest członkiem zespołu i ściśle z nim współpracuje.
Istota prac wstępnych	Przeprowadzane są szczegółowe badania rynku, planowanie produktu i analizy biznesowe	Ograniczają się do stworzenia wizji, która ogólnie opisuje wygląd i działanie produktu.
Zdefiniowanie produktu	Wymagania są określone i zatwierdzane w początkowej fazie	Produkt odkrywany jest stopniowo, a wymagania krystalizują się w trakcie
Odpowiedź zwrotna	Dostępna po wypuszczeniu produktu na rynek	Wczesna i częsta odpowiedź zwrotna po małych wdrożeniach

1.2. PODEJŚCIE KLASYCZNE

Podejście klasyczne, reprezentowane przez PMBoK (Kompendium wiedzy o zarządzaniu projektami) lub metodykę PRINCE (kompleksowa metoda zarządzania projektami, zalicza się ją do podejścia klasycznego) oraz jej następcę PRINCE2, ma na celu wytworzenie kompletnego produktu przy uprzednim, dokładnym określeniu jego cech. Takie podejście charakteryzuje się ogromnym formalizmem, weźmy na przykład dokonywanie zmian, które wiąże się z wypełnianiem dokumentów (RfC – ang. Request for Change) -prośby o zmianę. Każda odpowiedź, to z kolei oczekiwanie, aż zostanie przeanalizowana i zatwierdzona bądź odrzucona. Dodatkowo osoby odpowiedzialne zwykle nie pracują wraz z zespołem, w związku z czym często występują bariery komunikacyjne.[1]

1.3. PODEJŚCIE ZWINNE

Podejście zwinne ukierunkowane jest na zespół, który w pełni odpowiada za wykonanie swojej części zadania i stopniowo dostosowuje je do potrzeb przyszłych użytkowników. Przykładowe metodyki zwinne, to m.in.: Scrum, Lean, Cobit, SixSigma, Kanban, XP (ang. eXtream Programming), TDD (ang. Test-Driven Development) i FDD (ang. Feature-Driven Development). W praktycznej działalności często zespoły nie wykorzystują jednej metodyki, a opierają się na kilku. Przykładem może być tutaj niedawno powstały Scrum-ban, który jest połączeniem dobrych praktyk zaczerpniętych ze Scruma i Kanbana.[15] Ciekawym rozwiązaniem jest także TDD, czyli programowanie sterowane testami. Proponuje ono utworzenie przypadków testowych, zanim powstanie fragment kodu.[11] Każde z prezentowanych rozwiązań posiada swoje zalety, jednak najważniejsze jest dobranie odpowiedniej metodyki do realiów pracy i prawidłowa adaptacja względem realiów biznesowych, gdyż

ściśle stosowanie wszystkich praktyk może być nadmiernie pracochłonne w zastosowaniu do małych projektów.

1.4. MANIFEST AGILE

Manifest Agile powstał w 2001 roku, ale nie jest to sam początek tego ruchu zwinnego oprogramowania. Już wcześniej istniały pewne metodyki, jak również istniały podstawy teoretyczne dla wprowadzenia takich rozwiązań. Jeśli chodzi o podstawy teoretyczne, to trzeba zwrócić uwagę przede wszystkim na 3 kwestie:

- kwestię podejścia systemowego i szkoły systemowej (czyli lata 70-te XX w.), która dostarczyła dość znaczącej wiedzy pozwalającej na współczesne zarządzanie projektami;
- drugi aspekt, to zarządzanie jakością, koncepcje, metody zarządzania jakością, które są wykorzystywane bardzo mocno w metodykach zwinnych;
- trzeci aspekt, to zarządzanie wiedzą, czyli chociażby Takeuchi i Nonaka, którzy jako pierwsi wspomnieli o idei młyna (w artykule The new product development game, opublikowanym w „Harvard Business Review” w 1986 r.), skąd wzięła się później metoda „scrum”.

W latach 90-tych zaobserwowano znaczące skomplikowanie oprogramowania, tworzenia oprogramowania. Projekty dotychczas zarządzane klasycznie okazały się zbyt mało elastyczne, nie pozwalały na wystarczająco szybkie tworzenie oprogramowania. Zdecydowano więc, że trzeba jakoś zmodyfikować sposób, w jaki tworzymy oprogramowanie, aby odpowiadać na potrzeby klientów, na potrzeby rynku wystarczająco szybko. Pierwsze próby podjęto już na początku lat 90-tych XX w. Zostały one uwiecznione publikacją w 1995 r. metodyki scrum, opisu, w jaki sposób można stosować tą metodykę. Rok później pojawiła się Metodyka Extreme Programming, a więc już w połowie lat 90-tych mieliśmy metodyki zwinne, które stosowane były najpierw w ograniczonym zakresie, potem coraz szerzej. Również poszczególne organizacje, przedsiębiorstwa zaczęły tworzyć swoje odmiany tych metod. Tak więc dziś mamy całe bogactwo metodyk związanych z zarządzaniem zwinnym w projektach. Agile nie było zatem pierwsze, było po prostu pewnym podsumowaniem pierwszego etapu rozwoju tych metodyk zwinnych. W 2001 r. powstał Manifest, który określił, co jest ważne w zwinnym zarządzaniu projektem.1.1



Rys. 1.1. Manifesto for Agile Software Dev.[www.medium.com]

Ten Manifest pokazywał pewien system wartości, co jest ważniejsze, a co mniej ważne w zarządzaniu projektem. Mamy więc takie cztery porównania:

- Autorzy Manifestu twierdzili, że ludzie i interakcje między nimi są ważniejsi, niż procesy i narzędzia- to nie znaczy, że procesy i narzędzia nie są istotne, ale są mniej istotne, mniej ważne. Trzeba położyć większy nacisk na ludzi, na interakcje- to powoduje bardziej nieformalną komunikację, jej przyspieszenie, również przyspieszenie realizowania zadań i umożliwia bardziej elastyczne realizowanie tych zadań, kiedy zmieniają się warunki;
- Drugą zasadą jest orientacja bardziej na działające oprogramowanie, niż na dokumentację. Jeśli zerkniemy do starych wersji oprogramowania z lat 80-tych, 90-tych, do każdego programu dodawana była gruba instrukcja. Dzisiaj już o tym zapomnieliśmy. Dzisiaj wiele aplikacji nie ma w ogóle nawet instrukcji- mówimy, że działają intuicyjnie (przynajmniej powinny). Dzięki temu, że orientujemy się na realizację tych najważniejszych efektów w projekcie, możemy lepiej wykorzystać zasoby, możemy szybciej osiągnąć te efekty, a rzeczy mniej ważne, mniej istotne, takie właśnie, jak szczegółowa dokumentacja (jakaś dokumentacja przecież musi być), możemy odłożyć, możemy przeznaczyć dla nich mniejsze zasoby.
- Również jeśli chodzi o współpracę z klientem, w metodykach zwinnych proponuje się zmianę podejścia. Zamiast negocjować szczegółowo umowy, budujemy współpracę z tym klientem dlatego, że nie jesteśmy w stanie z góry przewidzieć, jaki będzie, tak do końca, zakres naszego projektu, co w tym projekcie zrealizujemy, co będzie potrzebne za rok, kiedy nasz produkt będzie prawie gotowy. Czy te wymagania się nie zmieniają wielokrotnie, biorąc pod uwagę szybkość zmiany technologii, potrzeb, oczekiwań klientów, szybkość zmian na rynku. Zatem klient powinien być blisko, powinien dostarczać bieżące informacje o swoich potrzebach, a w kontrakcie zawieramy tylko te informacje, które są najważniejsze.
- I w końcu reagowanie na zmiany zamiast szczegółowego planowania. Oczywiście

planowanie występuje w metodykach zwinnych, ale jest ono ograniczone tylko do tego, żeby dało się zarządzać takim projektem. Natomiast przede wszystkim orientujemy się na reagowanie na zmiany: zmiany potrzeb klienta, zmiany na rynku. Na dostosowanie naszego projektu, w kolejnych iteracjach, do tego, czego klient oczekuje.

Czasem niektórzy mówią, może żartobliwie, ale nieraz całkiem serio, że jeżeli czegoś nie zaplanowali, to stosowali właśnie Agile. Nic bardziej błędnego: w Agile każda iteracja jest planowana, w każdym dniu planujemy swoją pracę, stosujemy inne metody, rzadko stosujemy harmonogram Gantta, ale także planujemy te działania. Zatem taki polski Agile („polnische Agile”, jak niektórzy mówią), to przykład niewłaściwego zarządzania przedsięwzięciami i raczej nie należy się tym chwalić. Warto jednak zauważyć, że nie do każdego projektu możemy zastosować metodyki zwinne. One się lepiej sprawdzają wtedy, kiedy mamy:

- bardzo krótkie, napięte terminy;
- projekty mają charakter unikatowy;
- są skomplikowane.

Mamy do zrealizowania coś nowego, nieoczekiwanego i mało czasu. Wtedy ta metodyka zwinna rzeczywiście jest bardziej uzasadniona niż metodyki klasyczne. Stosowanie metodyk zwinnych, szczególnie żądanie tej interakcji między pracownikami ogranicza nam wielkość zespołu, a więc ogranicza nam wielkość projektu. Generalnie metodyki zwinne stosujemy:

- w małych i średnich projektach, rzadziej w projektach dużych;
- konieczne jest, aby w metodyce zwinnej dostępny był dla nas klient, klient musi się na bieżąco kontaktować z nami i mówić czego potrzebuje, jakie są jego oczekiwania, czy jest zadowolony z tego, co uzyskuje w poszczególnych iteracjach;
- tematyka projektu musi być taka, aby klient z każdej iteracji miał jakąś wartość, bowiem staramy się często wypuszczać oprogramowanie, często wprowadzać nowe jego wersje, ale to powoduje, że ta nowa wersja musi dostarczyć jakąś wartość dla klienta.

W przypadku oprogramowania jest to oczywiste. W przypadku, kiedy budujemy jakiś budynek, być może Agile wtedy nie jest aż tak przydatny. Trzeba się zastanowić, czy możemy zastosować całą metodykę Agile, czy jak współcześnie w wielu projektach, zastosować ją tylko w odniesieniu do wybranych modułów projektu, tam, gdzie rzeczywiście ma ona zastosowanie.[3]

2. NAJCZĘŚCIEJ WYKORZYSTYWANE NARZĘDZIA W ZARZADZANIU PROJEKTAMI.

Zarządzanie projektami informatycznymi ściśle wiąże się z wykorzystaniem narzędzi informatycznych, które wspomagają ten proces. Przy ich wyborze warto pamiętać, iż mają pomagać w pracy projektowej, a nie przeszkadzać w jej realizacji, dlatego należy wybierać je mądrze. Przykładem nieodpowiedniego doboru narzędzia może być sytuacja, w której kierownik projektu nie dotrzymuje terminów swoich prac, ze względu na zajmowanie się raportowaniem postępu prac lub aktualizacją harmonogramu.[8]

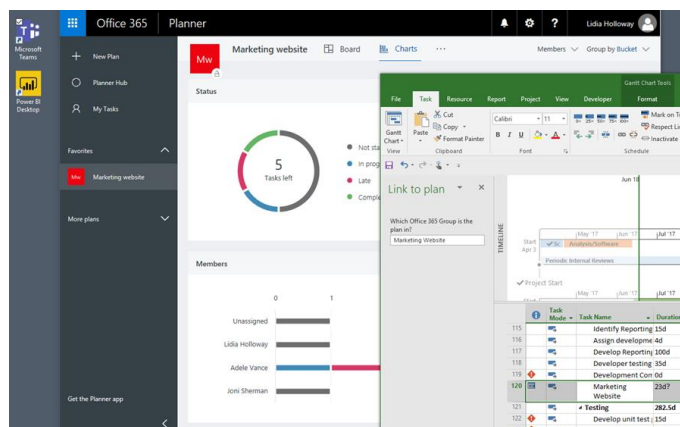
2.1. NARZĘDZIA STOSOWANE DLA METOD KLASYCZNYCH.

2.1.1. Microsoft Project

Najbardziej popularnym narzędziem stosowanym w metodykach klasycznych jest Microsoft Project. Pozwala on na rozpisanie całego harmonogramu działań, zaplanowanie budżetu czy stworzenie wykresu Gantta, tak niezbędnego w pracy kierownika. Pozwala także na tworzenie raportów, prezentacji i wykresów z postępów prac.[2.1]

2.1.2. Gantt Project

W niektórych przedsiębiorstwach w zarządzaniu projektami używa się programu Gantt Project. Jest to darmowe narzędzie umożliwiające dynamiczne tworzenie diagramów Gantta z podziałem na poszczególne zadania wraz z rozplanowaniem ich w czasie. Dodatkowo pozwala ono na tworzenie wykresów PERT (ang. Program Evaluation and Review Technique) wraz ze ścieżkami krytycznymi.[10]



Rys. 2.1. Microsoft Project[www.microsoft.com]

2.2. NAJPOPULARNIEJSZE NARZĘDZIA STOSOWANE DLA METOD ZWINNYCH (TEAM ESTIMATION GAME, KANBAN, PLANNING POKER).

W małych zespołach projektowych, które wykorzystują metodyki zwinne, odchodzi się zazwyczaj od Microsoft Project czy GanttProject stosując aplikacje webowe. Przykładem takich aplikacji są: Jira, Mantis, TFS, Trello, Sllack, KanbanTool. Przy metodykach zwinnych, właściciel produktu tworzy rejestr produktowy (ang. Product Backlog), który ma formę listy zawierającej wymagania klienta z określoną wagą (priorytetem) i czasochłonnością.[14] Wymienione wyżej narzędzia pozwalają na taką pracę i umieszczanie danych w chmurze, dzięki czemu każdy członek zespołu ma do nich dostęp, niezależnie od lokalizacji, czy pracuje w biurze czy poza nim. Zadania w backlogu¹ powinny być rozpisane dla obecnego sprintu² oraz zawierać dodatkowe zadania, które będą zasilać kolejny sprint bądź zostaną wykonane w istniejącym, gdy nadarzy się taka możliwość. Takie odejście do pracy umożliwia wykonanie części zadań przed wyznaczonym czasem. Narzędzia te pozwalają również na przypisanie konkretnego zadania do danego użytkownika, dzięki czemu każdy pracownik zna zakres swojej odpowiedzialności; takie rozwiązanie zapobiega wykonaniu tego samego zadania przez kilka osób. Niektóre firmy, a można nawet powiedzieć, że wiele firm, niezależnie od wykorzystywanej metodyki używa Microsoft Excela do zarządzania projektami (przyzwyczajenia?). Umożliwia on przedstawienie danych w postaci tabeli oraz różnych grafów. Każda firma może zarządzać nim na swój własny sposób, dzięki czemu proces ten jest bardzo elastyczny. Ponadto ułatwia on wykonanie prognoz przyszłych dochodów, kalkulacji wybranych parametrów i wskaźników. Kolejnym narzędziem wykorzystywanym na szeroką skalę jest SharePoint, który umożliwia współdzielenie plików czy tworzenie listy zadań, która może być wykorzystana w MS Project.

Przejdźmy wreszcie do narzędzi stosowanych w metodach zwinnych. Bardzo często

¹ Backlog – rejestr sprintu / lista zadań[6]

² Sprint – jeden z etapów niektórych metodyk zwinnych, który wyznacza rytm pracy. W jego trakcie następuje faktyczne wykonanie określonej funkcjonalności[9]

w projektach zwinnych planujemy bez użycia dni i godzin. W klasycznych projektach tworzymy harmonogramy, szczegółowe plany, gdzie mamy daty, gdzie mamy godziny, gdzie każde zadanie ma określony czas realizacji. W przypadku projektów zwinnych często odchodzimy od tak szczegółowego planowania, ale jakaś forma planowania oczywiście jest potrzebna, dlatego stosujemy różne alternatywne metody i jedną z takich alternatywnych metod jest metoda nazwana Team Estimation Game.

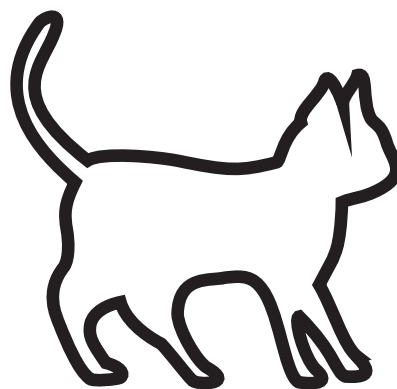
2.3. ODNIESIENIE DO KSIĄŻKI

Jak pisze Harel w [?]: Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

2.3.1. Rysunek z kotem

Jak widać na rys.2.2 Ala ma kota. Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetur a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetur. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetur odio sem sed wisi.



Rys. 2.2. Ala ma kota (opr.wł).

2.3.2. Tabela

Co uwzględniono w tabeli 2.1. Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula.

Tabela 2.1. Co kto ma [?] (patrz też dodatek A)

Ala	ma	kota
Ola	ma	psa
Ula	ma	małpę

Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

Warto wspomnieć, że w [2] rzecz przedstawiona jest zupełnie inaczej. Poniższy wzór:

$$\sum_{i=1}^{\infty} a_i \quad (2.1)$$

Wzór 2.1 wskazuje że dowód podany w [7] może zostać podważony. Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetur a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetur. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

2.4. LISTING

W moim kodzie 2.1 zrobiłem coś wspaniałego. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

```
int main()
{
    int a=2*3;
    printf("**Ala ma kota\n**");
    while(!I2C_CheckEvent(I2C1, I2C_EVENT_MASTER_MODE_SELECT)); /* EV5 */
    return 0;
}
```

Listing 2.1: Przykładowy algorytm w języku C (opr. wł.)

ZAKOŃCZENIE

W pracy udało mi się dużo zrobić. Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam congue neque id dolor.

Mnóstwo innych rzeczy da się poprawić i rozwinąć. Donec et nisl id sapien blandit mattis. Aenean dictum odio sit amet risus. Morbi purus. Nulla a est sit amet purus venenatis iaculis. Vivamus viverra purus vel magna. Donec in justo sed odio malesuada dapibus. Nunc ultrices aliquam nunc. Vivamus facilisis pellentesque velit. Nulla nunc velit, vulputate dapibus, vulputate id, mattis ac, justo. Nam mattis elit dapibus purus. Quisque enim risus, congue non, elementum ut, mattis quis, sem. Quisque elit.

Dodatki

A. TO POWINIEN BY? DODATEK

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetur a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetur. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetur odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetur eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

BIBLIOGRAFIA

- [1] 4pm, *Przyjrzyjmy się tradycyjnym projektom (porównanie prince2 oraz agilepm)*, <http://4pm.pl/artykuly/przyjrzyjmy-sie-tradycyjnym-projektom>. Ost. dost. 4 grudnia 2018.
- [2] Aizawa, K., Ogawa, Y., Ishido, T., *Groundwater flow and hydrothermal systems within volcanic edifices: Delineation by electric self-potential and magnetotellurics*, J. Geophys. Res. 2009, tom 114.
- [3] Cohen, M., *Agile Estimating And Planning*, Robert C. Martin Series (Pearson Education, Inc., Massachusetts, 2006).
- [4] Docker Inc., *Compose file version 3 reference*, <https://docs.docker.com/compose/compose-file/>. Ost. dost. 12 listopada 2018.
- [5] Institute, P.M., *A Guide to the Project Management Body of Knowledge* (Project Management Institute, Inc., Newton Square, Pennsylvania USA, 2000).
- [6] Jabłoński, B., *Wykorzystanie metodyk zwinnych do poprawy wiedzy i umiejętności projektowych studentów kierunków technicznych*, Edukacja - Technika - Informatyka. 2016, tom 94.
- [7] Kaleta, J., Lewandowski, D., Zajac, P., *Experimental identification of magnetorheological composites and elastomers properties*, w: *Materials Science Forum*, tom 482 (Aedermannsdorf, Switzerland: Trans Tech Publications, 1984-, 2005), str. 403–406.
- [8] Kopczewski, M., *Alfabet zarządzania projektami. Zarządzanie projektem od A do Z*, 2 wyd. (Helion, Massachusetts, 2015).
- [9] Ćwiklicki M., Jabłoński M., W.T., *Samoorganizacja w zarządzaniu projektami metodą Scrum* (Milfes.pl, wydawnictwo@milfes.pl, 2010).
- [10] Mariusz Sołtysik, M.W., *Współczesne trendy w zarządzaniu projektami* (Mfiles.pl, Kraków, 2016).
- [11] Percival, H.J., *Test Driven Development in Python* (Helion, ul. Kościuszki 1c, 44-100 Gliwice, 2015).
- [12] Portny, S.E., *Project Management For Dummies*, 3d edition wyd., For Dummies (Helion, ul. Kościuszki 1c, 44-100 Gliwice, 2013).
- [13] projektgamma, *Zespoły rozproszone*, <https://www.projektgamma.pl/strefa-wiedzy/wiki/zespoły-rozproszone>. Ost. dost. 4 grudnia 2018.
- [14] Shwaber, K., *Sprawne zarządzanie projektami metodą Scrum*, polish wyd. (APN PROMISE, Warszawa, 2005).
- [15] Wolf, H., *Zwinne projekty w klasycznej organizacji Scrum, Kanban, XP* (Helion, ul. Kościuszki 1c, 44-100 Gliwice, 2014).