

# Cloud Factory Studio

Dokumentacja projektowa

Aplikacja Analizująca Stan Fizyczny  
Użytkownika

# Health Together

Wersja 2.0

**Autorzy:**

Giełdowski Daniel

Kuc Piotr

Różański Antoni

Świerczek Arkadiusz

Warszawa, 31 Maja 2017

# Spis treści

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Wstęp</b>  | <b>3</b>  |
| <b>2</b> | <b>Zakres projektu</b>  | <b>4</b>  |
| 2.1      | Aspekt Biznesowy . . . . .  | 4         |
| 2.2      | Aspekt Użytkowy . . . . .   | 4         |
| 2.3      | Aspekt Techniczny . . . . .   | 4         |
| <b>3</b> | <b>Organizacja projektu</b>   | <b>5</b>  |
| 3.1      | Komitety sterujący . . . . .  | 5         |
| 3.2      | Zespoły projektowe . . . . .  | 5         |
| 3.2.1    | Lider zespołu . . . . .   | 5         |
| 3.2.2    | Członek zespołu . . . . .   | 5         |
| 3.2.3    | Zespół analityków . . . . .   | 5         |
| 3.2.4    | Zespół developerów . . . . .  | 6         |
| 3.2.5    | Zespół testerów . . . . .   | 6         |
| 3.2.6    | Zespół wsparcia . . . . .   | 6         |
| 3.2.7    | Zespół administratorów . . . . .                                      | 6         |
| 3.3      | Kierownik projektu . . . . .  | 6         |
| <b>4</b> | <b>Procedury sterowania i kontroli projektu</b>                       | <b>7</b>  |
| 4.1      | Procedury komunikacji . . . . .                                       | 7         |
| 4.1.1    | Procedury komunikacji wewnątrz firmy . . . . .                        | 7         |
| 4.1.2    | Procedury komunikacji z projektodawcą . . . . .                       | 8         |
| 4.1.3    | Procedury komunikacji z użytkownikiem aplikacji . . . . .             | 8         |
| 4.2      | Procedury zapewnienia i kontroli jakości . . . . .                    | 9         |
| 4.2.1    | Ustalenie oczekiwanych norm jakości . . . . .                         | 9         |
| 4.2.2    | Zapewnienie jakości . . . . .   | 9         |
| 4.2.3    | Kontrola jakości . . . . .  | 10        |
| 4.3      | Kontrola zgodności postępów prac oraz wykorzystania budżetu . . . . . | 11        |
| 4.4      | Kontrola zmian . . . . .  | 11        |
| 4.4.1    | Procedury kontroli zmian . . . . .                                    | 12        |
| 4.5      | Rozwiązywanie problemów . . . . .                                     | 13        |
| <b>5</b> | <b>Plan projektu</b>  | <b>14</b> |
| 5.1      | Etapy projektu . . . . .  | 14        |
| 5.1.1    | Uzgodnienie wymagań . . . . .   | 14        |
| 5.1.2    | Analiza projektu . . . . .  | 14        |
| 5.1.3    | Projektowanie i implementacja . . . . .                               | 14        |
| 5.1.4    | Testowanie . . . . .  | 15        |
| 5.1.5    | Podsumowanie i odbiór . . . . .                                       | 15        |
| 5.1.6    | Utrzymanie i wsparcie . . . . .                                       | 15        |
| 5.2      | Harmonogram trwania etapów . . . . .                                  | 15        |
| 5.3      | Harmonogram aktywności zespołów . . . . .                             | 15        |

|          |  |           |
|----------|--|-----------|
| <b>6</b> | <b>Analiza ryzyka</b>                            | <b>16</b> |
| 6.1      | Spis znanych ryzyk . . . . .                     | 16        |
| 6.2      | Macierz ryzyk . . . . .                          | 17        |
| 6.3      | Zdefiniowanie prawdopodobnych zagrożeń . . . . . | 18        |
| <b>7</b> | <b>Słownik użytych pojęć</b>                     | <b>18</b> |

| Data Modyfikacji | Numer Wersji | Opis Modyfikacji                                 |
|------------------|--------------|--|
| 09.05.2017       | 0.1          | Utworzenie dokumentacji                          |
| 12.05.2017       | 0.2          | Wstęp i zakres projektu                          |
| 13.05.2017       | 0.3          | Organizacja projektu                             |
| 15.05.2017       | 0.4          | Procedury sterowania i kontroli                  |
| 16.05.2017       | 1.0          | Dokumentacja wstępna                             |
| 20.05.2017       | 1.1          | Słownik pojęć,<br>Poprawki: Organizacja projektu |
| 26.05.2017       | 1.2          | Analiza ryzyka                                   |
| 28.05.2017       | 1.3          | Plan i harmonogram projektu                      |
| 31.05.2017       | 2.0          | Wersja ostateczna                                |

# 1 Wstęp

Celem dokumentu jest przedstawienie zarządzania realizacją projektu aplikacji "Health Together" dla firmy Capple. Firma Capple zajmuje się dystrybucją wysokiej jakości aplikacji na platformę Apple Watch. Wraz z rozrostem firmy niemożliwym stała się samodzielna realizacja wszystkich projektów i powstało zapotrzebowanie na zlecenie ich rozwijania podwykonawcom.

Z powodów opisanych powyżej podjęta została decyzja o realizacji aplikacji "Health Together" dla firmy Capple przez firmę Cloud Factory Studio. Aplikacja zrealizowana zostanie zgodnie z wszystkimi wymaganiami biznesowymi. Będzie ona wygodna zarówno dla klientów, umożliwiając dostęp do rzetelnych danych o ich stanie fizycznym, jak i pracowników pracujących w help-desku, udostępniając im stosowny interfejs.

Aplikacja spełniać będzie wszystkie wymagania biznesowe związane z prawem odnośnie prywatności.

## 2 Zakres projektu

### 2.1 Aspekt Biznesowy

Dzięki wprowadzeniu na rynek aplikacji "Health Together" dla firmy Capple możliwe będzie osiągnięcie poniższych możliwości biznesowych:

- Zwiększenie popularności firmy, dzięki wprowadzeniu nowatorskiej aplikacji.
- Powiększenie przewagi rynkowej nad konkurencją, dzięki zapełnieniu dotąd niezaspokojonej niszy rynkowej.

### 2.2 Aspekt Użytkowy

Dla końcowego użytkownika aplikacji aplikacja daje następujące korzyści:

- Otrzymywanie rzetelnych informacji na temat swojego aktualnego stanu fizycznego.
- Duża wygoda korzystania i mobilność umożliwiającą kontrolę zdrowia w każdej chwili.
- Wykrywanie stanów zagrażających zdrowiu i życiu oraz alarmowanie odpowiednich służb.
- Dostęp do najnowszych algorytmów obliczania i aproksymowania wskaźników zdrowia pacjenta.

### 2.3 Aspekt Techniczny

System oparty będzie na następującym modelu:

- Aplikacja stanowiąca interfejs wizualny na platformę Apple Watch korzystająca z udostępnionych przez urządzenie sposobów zbierania danych biometrycznych zaprogramowana za pomocą języka Swift z wykorzystaniem sieci neuronowych do analizy danych.
- Zewnętrzna baza danych zawierająca zebrane przez aplikację informacje.
- Komunikacja z bazą i alarmowanie zrealizowane za pomocą komunikacji urządzenia z siecią internetową (za pośrednictwem Wi-fi) oraz z siecią komórkową (za pośrednictwem karty SIM).

## 3 Organizacja projektu

### 3.1 Komitet sterujący

Celem komitetu sterującego jest czuwanie nad kierunkiem rozwoju projektu oraz podejmowanie najważniejszych decyzji biznesowych, które mają bezpośredni wpływ na produkt. Podczas każdego etapu projektu, osoby należące do komitetu powinny nadzorować postępy i dotychczasowy progres. Komitet powołuje oraz bezpośrednio nadzoruje kierownika projektu.

| Stanowisko            | Pracownik                   | Obowiązki  |
|-----------------------|-----------------------------|--|
| Przewodniczący        | Adam Michulski              | <ul style="list-style-type: none"><li>• Zwoływanie zebrania komitetu</li><li>• Podejmowanie kluczowych decyzji sterujących w sytuacjach kryzysowych</li></ul>  |
| Dyrektor techniczny   | Zbigniew Gorączka           | <ul style="list-style-type: none"><li>• Koordynacja architektury systemu i jego jakości</li><li>• Kontrola zgodności projektu ze wszelkimi normami oraz certyfikatami bezpieczeństwa</li><li>• Podejmowanie innych kluczowych decyzji technicznych</li></ul> |
| Konsultant techniczny | Przedstawiciel firmy Capple | <ul style="list-style-type: none"><li>• Wykonywanie niezależnego audytu i ekspertyz projektu realizowanego systemu</li><li>• Wyjaśnianie kwestii spornych i użyteczności odpowiednich technologii</li></ul>  |

### 3.2 Zespoły projektowe

#### 3.2.1 Lider zespołu

Każdy zespół składa się z maksymalnie 6 pracowników. Każdy zespół posiada lidera, który reprezentuje zespół przed kierownikiem projektu. Wszelka komunikacja między zespołami odbywa się za pomocą liderów. Lider powinien być uznanym autorytetem wśród wszystkich członków zespołu oraz powinien posiadać największe doświadczenie.

#### 3.2.2 Członek zespołu

Od członków poszczególnych zespołów wymaga się dobrej znajomości dziedziny, w której się specjalizuje. Z założenia członek zespołu musi pracować w małym zespole i raportować postęp przed liderem zespołu.

#### 3.2.3 Zespół analityków

Przygotowanie analizy procesów, wymagań, kosztów i interfejsów użytkownika. Do obowiązków członków tego zespołu będzie należało tworzenie dokumentacji biznesowej, wspomaganie programistów oraz przygotowanie planów testu aplikacji.

### 3.2.4 Zespół developerów

Zespół ten wykorzystując sprawdzone technologie zaprojektuje oraz wykona aplikację na inteligentny zegarek. Zakłada się, że dostępna będzie gotowa dokumentacja dostępnych czujników przygotowana przez twórców inteligentnego zegarka. Od deweloperów oczekuje się doskonałej znajomości języka Swift oraz technologii chmurowych oraz podstawowej wiedzy w zakresie sztucznych sieci neuronowych. Przed członkami zostało postawione zadanie przygotowania dodatkowo serwera zbierającego oraz analizującego dostępne dane.

### 3.2.5 Zespół testerów

Zespół testerów zostanie podzielony na dwa mniejsze podzespoły. Jeden będzie odpowiedzialny za bezpośrednie testowanie modułów zgodnie z wcześniej przygotowaną dokumentacją, a drugi za przygotowanie suity testowej - testów jednostkowych, end2end, integracyjnych jak i wydajnościowych. Pracownicy będą mieli za zadanie wychwytywać i zgłaszać zaistniałe problemy oraz błędy w działaniu aplikacji bezpośrednio do zespołu developerów.

### 3.2.6 Zespół wsparcia

Aby firma sprawnie działała, potrzebny jest również zespół wsparcia. Jego członkowie odpowiedzialni są za sprawy biurowe, księgowe i organizacyjne. Sprawują opiekę nad systemem Mantis. Pilnują przestrzegania standardów w firmie. Do ich zadań należy także komunikacja z klientem i umawianie dodatkowych spotkań, jak i komunikacja z użytkownikiem końcowym aplikacji.

### 3.2.7 Zespół administratorów

Projekt zakłada wymianę danych między użytkownikami, a serwerem. W tym celu należy powołać zespół administratorów zaznajomiony z zaawansowanymi zasadami bezpieczeństwa. W grę będą wchodzić dane osobowe klientów, których strata może narazić firmy na wielopoziomowe straty. Do obowiązków administratorów będzie należało zarządzanie serwerami z danymi oraz dbanie o bezpieczeństwo informacji.

## 3.3 Kierownik projektu

Kierownikiem projektu zostaje osoba wyznaczona przez komitet sterujący. Do jego obowiązków należy bezpośrednie kontrolowanie projektu. Od kierownika wymaga się podejmowania trafnych decyzji kierowniczych, dobrą znajomość możliwości używanych technologii oraz posiadania umiejętności zarządzania zasobami ludzkimi.

| Stanowisko                    | Pracownik            |
|-------------------------------|----------------------|
| Kierownik projektu            | Marysia Wajs         |
| Lider zespołu analityków      | Elżbieta Staropolska |
| Lider zespołu developerów     | Michał Jaworski      |
| Lider zespołu testerów        | Mikołaj Galopski     |
| Lider zespołu wsparcia        | Dominik Ziemniak     |
| Lider zespołu administratorów | Paweł Kupski         |

## 4 Procedury sterowania i kontroli projektu

### 4.1 Procedury komunikacji

#### 4.1.1 Procedury komunikacji wewnątrz firmy

##### 4.1.1.1 Komunikacja elektroniczna

Komunikacja wewnątrz firmy odbywa się za pośrednictwem platformy Slack - narzędzia do wygodnej współpracy zespołowej. Każdy zespół i dział ma swój dedykowany kanał, pozwalający na niezakłóconą komunikację i szybką reakcję na zaistniałe problemy. Oprócz tego istnieje szereg ogólnych kanałów firmowych, dzięki którym każdy upoważniony do takich informacji pracownik może zorientować się, na jakim etapie znajdują się prace innego zespołu. Konto każdego pracownika zostaje stworzone i zweryfikowane przez członka zespołu administratorów przed dołączeniem danego użytkownika do kanału, a wszelkie deklaracje oraz decyzje wyrażane w tym kanale komunikacji są traktowane jako jednoznaczne stanowisko nadawcy wiadomości. Również administratorzy odpowiadają za przydzielanie uprawnień do dołączania do istniejących kanałów firmowych.

Platforma Slack udostępnia swoje aplikacje na wiele platform, co umożliwia komunikację także w przypadku, gdy danego pracownika nie ma w danym momencie w biurze.

##### 4.1.1.2 Komunikacja interpersonalna

W każdym momencie dnia, jeśli zaistnieje taka potrzeba, kierownik zespołu może zarządzić indywidualne spotkanie z wybranym pracownikiem w celu rozwiązania zauważonych przez niego problemów.

Raz w tygodniu, lider każdego zespołu organizuje spotkanie dla swoich bezpośrednich podwładnych w salach konferencyjnych. Termin takiego spotkania jest ustalany wewnątrz zespołu. Spotkania te mają na celu raportowanie o aktualnym stanie prac, planowanie pracy w najbliższych dniach a także rozwiązywanie problemów na poziomie zespołowym.

Prace na najbliższy okres są przydzielane w wyniku dyskusji wewnątrz zespołu, w której decydujący głos ma kierownik zespołu.

Liderzy zespołów spotykają się raz w tygodniu, w terminie wstępnie ustalonym na czwartek, godz. 16.00, z Kierownikiem Projektu. Raportowane są na tym spotkaniu postępy prac każdego z zespołów oraz całego projektu. Analizowane są ryzyka i nowe wymagania projektodawcy, przeprowadzana jest również analiza obecnego stanu zasobów i ich planowanego zużycia w najbliższej przyszłości, w wyniku której ustalana jest strategia na najbliższy tydzień. Spotkania te mają charakter oficjalny i z każdego spotkania powstaje krótki dokument streszczający jego przebieg i wypunktowujący powzięte decyzje.

Przewodniczący Komitetu Sterującego zarządza spotkanie Komitetu z Kierownikiem Projektu oraz Pełnomocnikiem do spraw jakości nie rzadziej niż raz na 6 tygodni. Na spotkaniu Komitet Sterujący otrzymuje informacje o aktualnym stanie prac, zużyciu zasobów, napotkanych problemach i planowanych działaniach w najbliższej przyszłości. Pełnomocnik do spraw jakości przedstawia dokumenty raportujące kolejne kontrole jakości. Jeżeli zużycie zasobów, jakość wytworzonego produktu lub stan prac nie pokrywa się z tym przewidzianym w Harmonogramie Prac, Komitet Sterujący może powziąć wszelkie decyzje, jakie uważa za słuszne w celu poprawy sytuacji.



## 4.1.2 Procedury komunikacji z projektodawcą

### 4.1.2.1 Komunikacja elektroniczna

Członkowie zespołu wsparcia (helpdesku) odpowiedzialni są za kontakt z klientem; każdy z nich posiada swój osobisty adres e-mail w domenie firmowej. Mail ten ma przykładową postać *akowalczyk@healthtogether.com.pl*. Oprócz tego każdy z tej grupy pracowników ma swój służbowy telefon. Poprzez te dwa kanały komunikacji klient może skontaktować się z firmą w dowolnym momencie podczas godzin pracy. Firma uzgadnia z klientem maksymalny czas na odpowiedź, wynoszący w przypadku wiadomości e-mail 5 dni roboczych. Każdy następny dzień zwłoki, jeśli omawiana sprawa dotyczy krytycznych kwestii dla projektu, oznaczał będzie opóźnienie prac nad projektem nie z winy nadawcy wiadomości i strona opieszła w udzielaniu odpowiedzi będzie musiała liczyć się z poniesieniem kosztów tego opóźnienia uwzględnionymi w zawieranej umowie.

### 4.1.2.2 Komunikacja interpersonalna

W skład Komitetu Sterującego wchodzi także przedstawiciel zleceniodawcy. Jest on obecny na spotkaniach tej grupy i uczestniczy w podejmowaniu taktycznych decyzji odnośnie projektu. Oprócz tego, ustanawiane zostaną podczas akceptacji planu prac terminy spotkań Kierownika Projektu z kontrolerami jakości klienta. Spotkania te będą odbywać się nie częściej niż raz na miesiąc, a ich celem jest zaprezentowanie na żywo aktualnych prac nad aplikacją i dostosowanie przyszłych zmian do sugestii klienta.

### 4.1.2.3 Zgłaszanie postulatów i błędów

W trakcie trwania projektu, oprócz bezpośredniej komunikacji, klient może zgłaszać nowe postulaty dotyczące niektórych funkcji. W późniejszej fazie projektu, gdy zostanie udostępniona klientowi wersja alfa aplikacji, będzie mógł on też zgłaszać znalezione błędy.

Aby zapewnić profesjonalną i wygodną dla obu stron obsługę błędów i postulatów, zostanie uruchomiony firmowy *Mantis* - system zgłaszania i śledzenia błędów (ang. *bug tracker*). Za pośrednictwem tej platformy użytkownik będzie mógł zgłaszać postulaty, śledzić postępy aktualnie otworzonych zgłoszeń jak i ich historię czy powiązania między nimi. Pracownicy zespołu wsparcia, odpowiedzialni za opiekę nad Mantisem, będą między innymi akceptować zgłoszenia, przypisywać odpowiednim zespołom jak i je zamykać, gdy zostaną rozwiązane.

## 4.1.3 Procedury komunikacji z użytkownikiem aplikacji

Gdy aplikacja wejdzie w fazę beta, zostanie udostępniona użytkownikom - posiadaczom smartwatcha *Apple Watch*. Od tego momentu, do zadań zespołu deweloperów będzie należało utrzymanie aplikacji, a w tym także rozwiązywanie problemów zgłoszonych przez użytkowników dotyczących działania aplikacji na konkretnych wersjach systemu. Aplikacja będzie dostępna przez aplikację *Caplle Store*, za pomocą której użytkownicy końcowi będą mogli zarówno pobrać najnowszą wersję aplikacji, jak i ją oceniać i zgłaszać poprawki. Do obowiązków zespołu wsparcia (helpdesku) zaliczać będzie się również komunikacja z posiadaczami smartwatcha przez platformę *Apple Watch*.

## 4.2 Procedury zapewnienia i kontroli jakości

### 4.2.1 Ustalenie oczekiwanych norm jakości

Przed rozpoczęciem prac przygotowywany jest niniejszy Plan Realizacji projektu, którego zawartość jest uzgadniana z projektodawcą. Dołączany do tego dokumentu jest Harmonogram Prac z terminami i oczekiwanymi kosztami kolejnych etapów Projektu. Oddzielnie tworzony jest również dokument definiujący odpowiedzialności i kary podmiotów: zlecającego wykonanie (zleceniobiorcy, firmy Capple) i wykonującego Projekt (Cloud Factory Studio). Wszystkie dokumenty muszą zostać podpisane przez upoważnionych do takiej czynności przedstawicieli firmy Capple oraz Cloud Factory Studio przed początkiem prac i są punktem odniesienia w razie wystąpienia spornych kwestii podczas realizacji Projektu.

### 4.2.2 Zapewnienie jakości

Aby zapewnić jak najlepszą jakość wytwarzanego oprogramowania, zostaną zaimplementowane następujące elementy środowiska pracy:

#### 4.2.2.1 Stacje robocze

Każde ze stanowisk będzie wyposażone w stację roboczą umożliwiającą komfortową pracę. W parze z wysoką jakością podzespołów komputerowych będzie szła również wygoda użytkowania zapewniona przez minimum dwa monitory o przekątnej ekranu 27" dla każdego pracownika oraz indywidualnie wybierany fotel z oferty na stronie *foteleDlaBiur.com*.

Aby zapewnić bezawaryjność stacji roboczych, ich użytkownicy nie będą mieli możliwości samodzielnej instalacji oprogramowania. Jeśli których z pracowników miałby personalne preferencje co do programów, których chce lub potrzebuje używać, musi zgłosić to administratorowi przez odpowiedni kanał Slacka, lub, jeśli dotyczy to większej części firmy, przez prywatny firmowy dział Mantisa.

Firma zapewnia swoim pracownikom wykupione licencje na powszechnie używane aplikacje, takie jak Microsoft 365, zestaw narzędzi deweloperskich firmy JetBrains czy WinRAR.

#### 4.2.2.2 Środowisko pracy

Poszczególne zespoły będą pracowały w niedalekiej odległości od siebie, przy czym w żadnym pomieszczeniu nie będzie więcej jak 4 stanowiska pracy. Pomieszczenia są klimatyzowane, przestronne i mają dostęp do wielu źródeł światła. Każdy dział w firmie posiada osobną kuchnię i łazienkę. Kuchnia umożliwia przechowywanie i podgrzewanie posiłków, a firma zapewnia codziennie napoje i świeże owoce. Oprócz tego, do wspólnego użytku wewnątrzfirmowego zostanie udostępniony pokój do odpoczynku - z miejscem do rozmów oraz gier na konsoli.

#### 4.2.2.3 Infrastruktura

Aby maksymalizować produktywność, firma musi mieć ciągły dostęp do sieci oraz baz danych. Bezawaryjność systemu jest zapewniona przez codzienne kontrole (godz 19.30)

infrastruktury firmy przez zespół administratorów oraz ich regularne spotkania (raz na tydzień, termin ustalany przez lidera zespołu administratorów) w celu zapewnienia przejrzystej i wydajnej architektury sieci i serwerów.

#### 4.2.2.4 Kompetencje pracowników

Wszyscy pracownicy wyrażający taką chęć, mogą wziąć udział w bezpłatnym szkoleniu z używania platformy Slack, Mantis oraz używanego wewnątrz firmy oprogramowania CRM, aby poruszać się swobodnie i efektywnie w tych środowiskach. Oprócz tego, co miesiąc, w poniedziałki, będą odbywały się szkolenia z oprogramowania smartwatchy dla programistów i sztucznych sieci neuronowych oraz z kursy rozwijające umiejętności miękkie dla osób odpowiedzialnych za kontakt z klientem.

#### 4.2.3 Kontrola jakości

Zarząd firmy powołuje Pełnomocnika do Spraw Jakości. Jest on odpowiedzialny za kontrolę jakości wytwarzanego projektu informatycznego. Tworzy dokument - Plan jakości, w którym określa oczekiwane jakościowe rezultaty pracy firmy oraz sposób nadzoru nad nimi. Pełnomocnika do Spraw Jakości ma wgląd do wszelkich dokumentów oraz kodu oraz uczestniczy w spotkaniach Kierownika Projektu z Zarządem firmy, na których to spotkaniach prezentuje dokument będący efektem kontroli następujących kwestii:

1. Jakość kodu;
2. Jakość dokumentacji;
3. Zgodność postępów prac z Harmonogramem Prac;
4. Zgodność wykorzystania zasobów z planem;
5. Przestrzeganie standardów;
6. Wynik audytów;
7. Ocenę realizacji polityki jakościowej firmy

##### 4.2.3.1 Kontrola jakości kodu

Zespół deweloperów ma obowiązek tworzenia kodu wysokiej jakości, natomiast zadaniem zespołu testerów jest stworzenie takiego zestawu testów, który pokryje wszystkie możliwe przypadki. Będą tworzone testy:

1. Testy jednostkowe;
2. Testy integracyjne;
3. Testy systemowe;
4. Testy wydajnościowe.

Pełnomocnik do Spraw Jakości ocenia kod obu tych grup. Uwzględnia w swoim raporcie następujące czynniki:

1. Czytelność kodu;
2. Podział kodu na moduły i pakiety;
3. Pokrycie testów jednostkowych.

Aby umożliwić łatwą współpracę programistów oraz kontrolę kodu, utworzone zotaje repozytorium GIT pod adresem <https://github.com/CloudFactoryStudio/HealthTogether>.

Oprócz tego firma korzysta z narzędzia ciągłej integracji (Continuous Integration): Jenkins, który w na bieżąco informuje o stanie kodu w repozytorium.

#### 4.2.3.2 Kontrola jakości dokumentacji

Deweloperzy mają obowiązek tworzenia, obok nowych modułów, dokumentacji opisujących aspekty techniczne i użytkowe utworzonych modułów. Pełnomocnik do Spraw Jakości sprawdza zgodność dokumentacji analitycznej i technicznej ze stanem rzeczywistym, jak i jej transparentność i dokładność.

### 4.3 Kontrola zgodności postępów prac oraz wykorzystania budżetu

Obowiązek kontroli zgodności postępów prac oraz wykorzystania budżetu przypada Kierownikowi Projektu. Głównym narzędziem, które wykorzystuje podczas kontroli jest wewnętrzny firmowy system CRM. Każdy pracownik ma w nim swoje konto (zakładane przez zespół administratorów) oraz na bieżąco raportuje bieżące prace; zapisywana w bazie danych jest informacja jakie zadanie pracownik wykonywał danego dnia wraz z ilością czasu, jakie dane zadanie pochłonęło. Wszelkie dodatkowe wydatki, poniesione przez firmę, takie jak wydatki na infrastrukturę firmy zgłaszane przez administratorów, również są w tym programie zapisywane. Kierownik Projektu eksportuje z programu raporty miesięczne i tygodniowe, na podstawie których ocenia, czy postępy i wydatki mieszczą się w ich zaplanowanych wielkościach.

W przypadku stwierdzenia nieprawidłowości, Kierownik Projektu, po konsultacji z Komitetem Sterującym, może podjąć decyzję o jednej lub wielu następujących czynnościach:

1. Przesunięcia personalne w ramach zespołów
2. Zmniejszenie zakresu prac
3. Wprowadzenie nadgodzin i skrócenie urlopów

Do danych uzyskanych z systemu CRM dostęp ma również zleceniodawca.

### 4.4 Kontrola zmian

Dzięki systemowi zgłaszania i śledzenia błędów Mantis klient oraz pracownicy firmy mogą w szybki sposób reagować na zmianę wymagań. Aby ograniczyć niekorzystny wpływ zmian w projekcie, nakłada się obowiązek natychmiastowego poinformowania Kierownika Projektu, bezpośrednio lub poprzez platformę Slack o zaistniałej sytuacji od razu po jej wykryciu przez pracowników zespołu wsparcie w firmie, którzy zajmują się monitorowaniem systemu Mantis.

#### 4.4.1 Procedury kontroli zmian

Wszelkie zmiany muszą być udokumentowane. Ewidencję zmian prowadzi Kierownik Projektu, który na ich podstawie tworzy dokument, który jest dogłębnie analizowany podczas spotkań Komitetu i wtedy też jest ustalana odpowiednia reakcja na te zmiany.

Zmiany w projekcie mogą wynikać z:

1. Zmiany wymagań funkcjonalnych zlecniodawcy;
2. Zmiany wymagań technologicznych zlecniodawcy;
3. Błędów w specyfikacjach i dokumentacjach;
4. Nowych uregulowań prawnych i gospodarczych;
5. Nagłych zmian w dyspozycyjności pracowników;
6. Nagłych zmian w dostępności sprzętu i technologii;
7. Nagłych zmian w materialnych i sprzętowych zasobach firmy;
8. Naruszenia obowiązujących standardów.

Kierownik na podstawie każdej zmiany przedstawia Komitetowi dokument zawierający:

1. Datę zgłoszenia konieczności zmiany;
2. Przyczynę konieczności zmiany;
3. Oczekiwany rezultat zmiany;
4. Zgrubną kalkulację kosztu wprowadzenia zmiany;
5. Konsekwencje wprowadzenia zmiany na założone cele;
6. Plan wprowadzania zmiany.

Komitet sterujący po zapoznaniu się z dokumentem decyduje o koniecznych krokach które należy podjąć w celu minimalizacji kosztów wprowadzenia zmiany; w szczególności ocenia, czy zmiana jest konieczna, czy sposób wprowadzenia jej zaproponowany przez Kierownika spełnia założone wymogi jakości oraz budżetu. Przedstawiciel klienta może zmianę zaakceptować, odroczyć lub odrzucić.

Jeśli zmiana została zaakceptowana, Komitet wprowadza do planu wprowadzenia zmiany zaproponowanego przez Kierownika ewentualne korekty, po czym Kierownik przystępuje do realizacji planu. Zostaje przygotowany projekt wprowadzenia zmiany, który streszcza sposób jej wprowadzenia oraz testowania, razem z modułami powiązanymi.

## 4.5 Rozwiązywanie problemów

W firmie panują przejrzyste oraz powszechnie dostępne dla zainteresowanych zasady pracy. Każdy pracownik ma obowiązek zapoznać się z nimi i je zaakceptować. Zasady te narzucają pewne normy i ich przestrzeganie ma za zadanie pomóc w unikaniu konfliktów w firmie. Jeśli jednak takie wystąpią, istnieją procedury ich rozwiązywania, opisane już częściowo w sekcji 4.1.1.2. Za rozwiązywanie konfliktów na najniższym szczeblu odpowiedzialni są liderzy zespołów; muszą to być osoby potrafiące dostrzec sytuacje konfliktowe i umiejętnie je rozwiązać. W pierwszej kolejności odbywają się indywidualne rozmowy z pracownikiem którego dotyczy dana sytuacja. Jeśli jest to wymagane, problem jest poruszany w gronie zespołu na cotygodniowych spotkaniach raportujących stan prac. Jeśli problem wciąż nie udaje się rozwiązać, informowany jest o tym Kierownik Projektu, który podejmuje ostateczną decyzję dotyczącą istoty konfliktu.

Jeśli problem nie jest natury personalnej, lecz sprzętowej lub technicznej, pracownik może to zgłosić odpowiedniemu zespołowi poprzez platformę Mantis do użytku wewnątrz-firmowego.

## 5 Plan projektu

### 5.1 Etapy projektu

| Numer etapu | Nazwa etapu                   | Planowana data rozpoczęcia | Planowana data zakończenia | Zaangażowane osoby  |
|-------------|-------------------------------|----------------------------|----------------------------|---|
| 1.          | Uzgodnienie wymagań           | 05.06.2017                 | 12.06.2017                 | <ul style="list-style-type: none"> <li>• Komitet sterujący</li> <li>• Przedstawiciele klienta</li> <li>• Kierownik projektu</li> <li>• Zespół analityków</li> </ul> |
| 2.          | Analiza projektu              | 12.06.2017                 | 10.07.2017                 | <ul style="list-style-type: none"> <li>• Kierownik projektu</li> <li>• Zespół analityków</li> <li>• Zespół wsparcia</li> </ul>                                      |
| 3.          | Projektowanie i implementacja | 03.07.2017                 | 25.09.2017                 | <ul style="list-style-type: none"> <li>• Dyrektor techniczny</li> <li>• Kierownik projektu</li> <li>• Zespół developerów</li> <li>• Zespół wsparcia</li> </ul>      |
| 4.          | Testowanie                    | 28.08.2017                 | 02.10.2017                 | <ul style="list-style-type: none"> <li>• Konsultant techniczny</li> <li>• Kierownik projektu</li> <li>• Zespół testerów</li> <li>• Zespół wsparcia</li> </ul>       |
| 5.          | Podsumowanie i odbiór         | 02.10.2017                 | 09.10.2017                 | <ul style="list-style-type: none"> <li>• Komitet sterujący</li> <li>• Przedstawiciele klienta</li> <li>• Kierownik projektu</li> <li>• Zespół analityków</li> </ul> |
| 6.          | Utrzymanie i wsparcie         | 09.10.2017                 | 09.10.2022                 | <ul style="list-style-type: none"> <li>• Zespół administratorów</li> <li>• Zespół wsparcia</li> </ul>   |

#### 5.1.1 Uzgodnienie wymagań

Celem pierwszego etapu projektu jest precyzyjne określenie wymagań klienta. W czasie tego etapu bardzo ważna będzie współpraca z klientem w celu zebrania jak największej ilości informacji odnośnie planowanego wyglądu, działania i specyfikacji aplikacji.

#### 5.1.2 Analiza projektu

Podczas tego etapu zespół analityków ma za zadanie stworzyć na podstawie informacji zebranych od klienta wysokiej jakości dokumentację biznesową. W razie jakichkolwiek wątpliwości zespół powinien skontaktować się z klientem. Po ukończeniu analizy biznesowej zostanie stworzona także wysokiej jakości dokumentacja techniczna.

#### 5.1.3 Projektowanie i implementacja

W etapie tym nastąpi stworzenie szczegółowej dokumentacji technicznej oraz projektu aplikacji, a następnie jej implementacja. Etap zacznie się jeszcze na tydzień przed zakończeniem analizy projektu, kiedy gotowa będzie już analiza biznesowa i część analizy technicznej.

#### 5.1.4 Testowanie

Celem etapu będzie gruntowne przetestowanie działania aplikacji w celu wykrycia wszystkich błędów. Etap rozpocznie się na miesiąc przed zakończeniem poprzedniego w celu umożliwienia zespołowi developerów poprawy wykrytych nieprawidłowości. Efektem etapu będzie szczegółowy raport. Przeprowadzone zostaną także testy akceptacyjne w obecności konsultanta technicznego.

#### 5.1.5 Podsumowanie i odbiór

W tym etapie odbędzie spotkanie podsumowujące projekt oraz odebranie aplikacji przez klienta.

#### 5.1.6 Utrzymanie i wsparcie

Ostatnim etapem projektu będzie utrzymanie i konserwacja gotowej aplikacji, jej serwerów i baz danych, a także kontakt z użytkownikami w okresie 5 lat od oddania aplikacji.

### 5.2 Harmonogram trwania etapów

| Etap | Tygodnie  |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
|------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
|      | 05.<br>06 | 12.<br>06 | 19.<br>06 | 26.<br>06 | 03.<br>07 | 10.<br>07 | 17.<br>07 | 24.<br>07 | 31.<br>07 | 07.<br>08 | 14.<br>08 | 21.<br>08 | 28.<br>08 | 04.<br>09 | 11.<br>09 | 18.<br>09 | 25.<br>09 | 02.<br>10 | 09.<br>10 |
| 1.   |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| 2.   |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| 3.   |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| 4.   |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| 5.   |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| 6.   |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |

### 5.3 Harmonogram aktywności zespołów

| Zespół          | Tygodnie  |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
|-----------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
|                 | 05.<br>06 | 12.<br>06 | 19.<br>06 | 26.<br>06 | 03.<br>07 | 10.<br>07 | 17.<br>07 | 24.<br>07 | 31.<br>07 | 07.<br>08 | 14.<br>08 | 21.<br>08 | 28.<br>08 | 04.<br>09 | 11.<br>09 | 18.<br>09 | 25.<br>09 | 02.<br>10 | 09.<br>10 |
| Analitików      |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| Developerów     |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| Testerów        |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| Wsparcia        |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |
| Administratorów |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |           |



## 6 Analiza ryzyka

### 6.1 Spis znanych ryzyk

---

|                                 |   |
|---------------------------------|---|
| <b>Zagrożenie nr 1:</b>         | Zbyt optymistyczne planowanie   |
| Możliwe skutki:                 | <ul style="list-style-type: none"><li>• Niezrealizowanie planu w terminie</li><li>• Zwiększenie kosztów</li></ul> |
| Wpływ:                          | Duży  |
| Prawdopodobieństwo zaistnienia: | Niskie  |

---

|                                 |   |
|---------------------------------|---|
| <b>Zagrożenie nr 2:</b>         | Awaria chmury obliczeniowej   |
| Możliwe skutki:                 | <ul style="list-style-type: none"><li>• Brak możliwości użytkowania oprogramowania</li><li>• Straty finansowe</li></ul> |
| Wpływ:                          | Ogromny   |
| Prawdopodobieństwo zaistnienia: | Średnie   |

---

|                                 |  |
|---------------------------------|--|
| <b>Zagrożenie nr 3:</b>         | Błędy w oprogramowaniu   |
| Możliwe skutki:                 | <ul style="list-style-type: none"><li>• Wydanie wadliwego produktu</li><li>• Kary finansowe</li><li>• Opóźnienia w projekcie</li></ul> |
| Wpływ:                          | Ogromny  |
| Prawdopodobieństwo zaistnienia: | Średnie  |

---

|                                 |  |
|---------------------------------|--|
| <b>Zagrożenie nr 4:</b>         | Braki w zasobach ludzkich  |
| Możliwe skutki:                 | <ul style="list-style-type: none"><li>• Opóźnienia w projekcie</li><li>• Straty finansowe</li><li>• Wydanie wadliwego produktu</li></ul> |
| Wpływ:                          | Duży   |
| Prawdopodobieństwo zaistnienia: | Duże   |

---

|                                 |   |
|---------------------------------|---|
| <b>Zagrożenie nr 5:</b>         | Awaria sprzętu  |
| Możliwe skutki:                 | <ul style="list-style-type: none"><li>• Opóźnienia w projekcie</li><li>• Straty finansowe</li></ul> |
| Wpływ:                          | Duży  |
| Prawdopodobieństwo zaistnienia: | Niskie  |

|                                 |  |
|---------------------------------|--|
| <b>Zagrożenie nr 6:</b>         | Cyberatak  |
| Możliwe skutki:                 | <ul style="list-style-type: none"> <li>• Dostęp do poufnych informacji przez niepowołane osoby</li> <li>• Kary finansowe</li> <li>• Zmiana informacji w bazie danych</li> <li>• Wyłączenie chmury obliczeniowej</li> </ul> |
| Wpływ:                          | Ogromny  |
| Prawdopodobieństwo zaistnienia: | Niskie   |
| <b>Zagrożenie nr 7:</b>         | Sabotaż pracownika   |
| Możliwe skutki:                 | <ul style="list-style-type: none"> <li>• Wydanie wadliwego produktu</li> <li>• Wyłączenie chmury obliczeniowej</li> </ul>  |
| Wpływ:                          | Ogromny  |
| Prawdopodobieństwo zaistnienia: | Niskie   |
| <b>Zagrożenie nr 8:</b>         | Konflikty i nieporozumienie w zespole  |
| Możliwe skutki:                 | <ul style="list-style-type: none"> <li>• Wydanie wadliwego produktu</li> <li>• Opóźnienia</li> <li>• Zmiany organizacyjne w zespołach</li> </ul>   |
| Wpływ:                          | Średni   |
| Prawdopodobieństwo zaistnienia: | Średnie  |
| <b>Zagrożenie nr 9:</b>         | Niedostępność miejsca pracy  |
| Możliwe skutki:                 | <ul style="list-style-type: none"> <li>• Utrudniona komunikacja z innymi członkami projektu</li> </ul>   |
| Wpływ:                          | Niski  |
| Prawdopodobieństwo zaistnienia: | Niskie   |

## 6.2 Macierz ryzyk

|                            |              |       |        |      |         |
|----------------------------|--------------|-------|--------|------|---------|
| <b>Prawdopodobieństwo:</b> |              |       |        |      |         |
| Duże                       |              |       |        | 4    |         |
| Średnie                    |              |       | 8      |      | 3, 2    |
| Niskie                     |              | 9     |        | 5, 1 | 6, 7    |
| Bardzo niskie              |              |       |        |      |         |
| <b>Wpływ: :</b>            | Bardzo niski | Niski | Średni | Duży | Ogromny |

### 6.3 Zdefiniowanie prawdopodobnych zagrożeń

**Zagrożenie nr 2:** Awaria chmury obliczeniowej  
Działania prewencyjne:

- Ograniczenie ilości przetwarzanych danych do minimum
- Przeniesienie części obliczeń na inną chmurę

---

**Zagrożenie nr 3:** Błędy w oprogramowaniu  
Działania prewencyjne:

- Wydłużony czas na etap testowania
- Zatrudnienie większej ilości testerów
- Zwiększenie budżetu przeznaczonego na audyt kodu

---

**Zagrożenie nr 4:** Braki w zasobach ludzkich  
Działania prewencyjne:

- Umożliwienie pracy zdalnej
- Wdrożenie elastycznych godzin pracy
- Przyjęcie pewnego prawdopodobieństwa niedostępności pracowników podczas planowania
- Określenie dostępności innych pracowników i przydzielenie ich na czas określony do innego zespołu

## 7 Słownik użytych pojęć

| Termin    | Opis   |
|-----------|--|
| Klient    | Firma zamawiająca aplikację, "Capple"  |
| Pracownik | Osoba zatrudniona w firmie "Cloud Factory" biorąca udział w projekcie                                |
| Aplikacja | Tworzona aplikacja o nazwie "Health Together" przez firmę "Cloud Factory" na zlecenie firmy "Capple" |