**Skład grupy:**

Wiktor 13599 – lider

Daniel 13996 - projektant

Michał 10894 - analityk

Paweł 14073 - programista

Sebastian 13825 - programista

**Opis wyglądu aplikacji:**

Po włączeniu aplikacji otwiera się okno do logowania.

Kurier loguje się swoim indywidualnym ID który otrzymuję przez pracodawcę.

Przy pierwszym logowaniu kurier ustala swoje hasło, którym będzie musiał się logować przy otworzeniu aplikacji.

Dla wygody po ustawieniu hasła jest także możliwość ustawienia logowania przez skaner odcisków palców bądź skaner twarzy.

Po zalogowaniu się wyświetla się główne menu w którym do wyboru są poniższe kategorie:

* informacja odnośnie ilości paczek do wydania w dniu obecnym
* mapa w trybie live pokazująca obecne utrudnienia na drodze m.in. korki, roboty drogowe, najbliższe stacje paliw oraz restauracje w których kurier podczas wyznaczonych przerw może przebywać
* rozpoczęcie pracy/ rozpoczęcie rozwożenia paczek, po zaciągnięciu danych z bazy system ustala najszybszą trasę (aplikacja analizuje ruch w danym mieście, który także jest uwzględniany podczas tworzenia danej trasy) do rozwożenia poszczególnych paczek uwzględniając dostarczenie paczek bezpośrednio pod adres klienta/adres punktu odbioru (np. sklep, placówka) oraz paczkomat.
* uszkodzenie paczki bądź zagubienie podczas transportu, kurier po numerach paczki/danych klienta ma możliwość zgłoszenia w systemie danej paczki, która finalnie nie zostanie dostarczona
* AWARIA/PROBLEM możliwe jest zgłoszenie opóźnienia w danym dniu przez skutki losowe (np. pęknięcie opony, kolizja, awaria pojazdu oraz pole tekstowe do wpisania przez kuriera) następnie po stworzeniu zgłoszenia oddzwania pracownik biurowy z informacją co dalej z obecną sytuacją, ewentualnie w wyjątkowych przypadkach jest od razu możliwość wyboru kontaktu z przełożonym bądź osobą, która w danym obecnie dniu ma dyżur na bazie

**Harmonogram:**

**Faza 1: Planowanie, projektowanie i zakup licencji (6 dni, koszt: 28 000)**

Zebranie całego zespołu i omówienie aplikacji kurierskiej, stworzenie planu jej tworzenia i rozwoju **(Wiktor, Daniel, Michał, Sebastian, Paweł**):

* Zespół przeprowadzi burzę mózgów i omówi różne pomysły na aplikację, biorąc pod uwagę takie czynniki jak zapotrzebowanie klienta, wykonalność techniczną i potrzebne zasoby.
* Zespół sfinalizuje zakres i cele projektu oraz stworzy plan ich osiągnięcia.

Programiści i projektant tworzą wstępne projekty i przedstawiają plan działania na następnym spotkaniu (**Sebastian, Paweł, Daniel**):

* Programiści i projektant tworzą wstępne projekty aplikacji. (**Sebastian**)
* Projekty zostaną przedstawione reszcie zespołu na następnym spotkaniu w celu uzyskania informacji zwrotnej i dyskusji.
* Zespół wspólnie dopracuje projekty poprzez ustalenie planów dotyczących funkcjonalności i interfejsu aplikacji.

Spotkanie z klientem w celu omówienia wstępnych projektów (**Wiktor, Daniel**):

* Lider spotka się z klientem, aby przedstawić wstępne projekty aplikacji i zebrać informacje zwrotne.
* Projektant będzie obecny, aby przedstawić szczegóły dotyczące projektów i odpowiedzieć na wszelkie pytania, jakie może mieć klient.
* Lider wprowadzi wszelkie niezbędne zmiany do planu projektu w oparciu o informacje zwrotne od klienta.

Zakup niezbędnych licencji (**Wiktor**):

* Lider zidentyfikuje wszelkie niezbędne licencje, które są wymagane do rozwoju i wdrożenia aplikacji.
* Lider będzie negocjował z odpowiednimi stronami, aby uzyskać niezbędne licencje.
* Lider zapewni, że wszystkie wymagane licencje są odpowiednio pozyskane i udokumentowane.

**Faza 2: Wstępne tworzenie aplikacji (50 dni, koszt: 55 000)**

Programiści tworzący podstawy strony (podstawowe funkcje i wygląd zewnętrzny - FRONTEND) (**Sebastian, Paweł, Daniel**):

* Programiści w ścisłej współpracy z projektantem będą pracować nad stworzeniem podstawowych funkcji aplikacji i wyglądu zewnętrznego strony, w tym panelu logowania i menu głównego. (**Sebastian**)
* Programiści w ścisłej współpracy z projektantem utworzą dodatkowe strony logowania, w tym logowanie za pomocą skanera linii papilarnych oraz skanowania twarzy.
* Programiści w ścisłej współpracy z projektantem stworzą frontend do zgłaszania awarii/problemów oraz do zgłaszania problemów z paczkami (osobna zakładka niż awarie/problemy). (**Sebastian**)

Projektant przeglądający i poprawiający projekt (**Daniel**):

* Projektant dokona przeglądu projektu aplikacji i wprowadzi wszelkie niezbędne poprawki, aby zapewnić, że jest ona atrakcyjna wizualnie i przyjazna dla użytkownika.
* Projektant będzie ściśle współpracował z programistami, aby zapewnić prawidłową realizację projektu.

Testowanie i debugowanie (**Michał, Sebastian**):

* Analityk będzie współpracował z programistami w celu przetestowania i debugowania podstawowych funkcji i wyglądu zewnętrznego aplikacji.
* Analityk zapewni, że wszystkie aspekty aplikacji działają poprawnie.

Zespołowa prezentacja postępów (W**iktor, Daniel, Michał, Sebastian**):

* Lider zwoła spotkanie z projektantem, analitykiem oraz programistami, aby przedstawić dotychczasowe postępy w realizacji projektu.
* Zespół omówi stan realizacji projektu oraz wszelkie problemy i wyzwania, które się pojawiły.
* Zespół dokona wszelkich niezbędnych korekt w planie projektu w oparciu o dotychczasowe postępy.

Spotkanie z klientem w celu omówienia bieżącego projektu (**Wiktor**):

* Lider spotka się z klientem, aby omówić aktualny stan projektu i upewnić się, że potrzeby i oczekiwania klienta są spełniane.
* Lider przedstawi informacje na temat postępów w realizacji projektu i odpowie na wszelkie pytania i wątpliwości, jakie może mieć klient.
* Lider zbierze informacje zwrotne od klienta i dokona zmian w planach projektu.

Lider omawiający wynik spotkania z zespołem (**Wiktor, Daniel, Paweł, Michał, Sebastian**):

* Lider spotka się z zespołem, aby omówić wszelkie obszary, w których potrzebne są zmiany i poprawki na żądanie klienta.
* Lider będzie zabiegał o informacje zwrotne od reszty zespołu i rozważał ich sugestie dotyczące poprawek.
* Lider będzie współpracował z zespołem, aby wprowadzić wszelkie niezbędne poprawki do planu projektu w oparciu o otrzymane informacje zwrotne od nich i od klienta.

**Faza 3: Rozwój aplikacji (35 dni, koszt: 80 000)**

Programiści dodają kolejne funkcje do aplikacji (BACKEND) (**Sebastian, Paweł, Michał**):

* Programiści będą pracować nad dodaniem zaawansowanych funkcji do aplikacji, w tym backendu do zgłaszania awarii/problemów oraz backendu do zgłaszania problemów z paczkami.
* Programiści będą ściśle współpracować z analitykiem, aby zapewnić, że zaawansowane funkcje są odpowiednio zintegrowane z aplikacją. (**Sebastian**)

Testowanie i debugowanie wstępne (**Michał, Sebastian, Paweł**):

* Analityk będzie współpracował z programistami w ramach wstępnego testowania i debugowania funkcji aplikacji.
* Analityk zapewni, że wszystkie aspekty aplikacji działają poprawnie i są gotowe do następnej fazy rozwoju.

Lider przeglądający postępy i wprowadzający poprawki (**Wiktor**):

* Lider będzie dokonywał przeglądu postępów w realizacji aplikacji i wprowadzał wszelkie niezbędne korekty do planu projektu.
* Lider będzie współpracował z zespołem, aby zapewnić, że projekt jest na dobrej drodze i że wszystkie cele i założenia są realizowane.

Spotkanie z klientem w celu omówienia bieżącego projektu (**Wiktor**):

* Lider spotka się z klientem, aby omówić aktualny stan projektu i upewnić się, że potrzeby i oczekiwania klienta są spełniane.
* Lider przedstawi informacje na temat postępów w realizacji projektu i odpowie na wszelkie pytania i wątpliwości, jakie może mieć klient.
* Lider zbierze informacje zwrotne od klienta i dokona zmian w planach projektu.

Lider omawiający wynik spotkania z zespołem (**Wiktor, Daniel, Paweł, Michał, Sebastian**):

* Lider spotka się z zespołem, aby omówić wszelkie obszary, w których potrzebne są zmiany i poprawki na żądanie klienta.
* Lider będzie zabiegał o informacje zwrotne od reszty zespołu i rozważał ich sugestie dotyczące poprawek.
* Lider będzie współpracował z zespołem, aby wprowadzić wszelkie niezbędne poprawki do planu projektu w oparciu o otrzymane informacje zwrotne od nich i od klienta.

**Faza 4: Integracja i doskonalenie aplikacji (25 dni, koszt: 60 000)**

Programiści integrują zaawansowane funkcjonalności (**Sebastian, Paweł, Michał, Daniel**):

* Programiści będą pracować nad integracją zaawansowanych cech i funkcjonalności w aplikacji, między innymi tworzenie baz danych, dodanie połączenia z Google Maps oraz dodanie funkcje wyznaczania optymalnej trasy dla kuriera.
* Programiści będą również pracować nad stworzeniem backendu dla panelu logowania oraz logowanie za pomocą skanera linii papilarnych oraz skanowania twarzy.
* Programiści będą ściśle współpracować z analitykiem, aby zapewnić, że integracja jest bezproblemowa i że wszystkie aspekty aplikacji działają poprawnie.
* Programiści będą ściśle współpracować z projektantem, aby usprawnić wygląd wizualny aplikacji i poprawić doświadczenie użytkowników.

Testowanie i debugowanie (**Michał, Sebastian, Paweł**):

* Analityk będzie współpracował z programistami w celu przetestowania i debugowania integracji zaawansowanych funkcjonalności aplikacji.
* Analityk zapewni, że wszystkie aspekty aplikacji działają poprawnie i są gotowe.

Lider przeglądający postępy i wprowadzający poprawki (**Wiktor**):

* Lider wraz z projektantem będzie dokonywał przeglądu postępów w realizacji projektu i wprowadzał wszelkie niezbędne korekty do planu projektu.
* Lider będzie współpracował z zespołem, aby zapewnić, że projekt jest na dobrej drodze i że wszystkie cele i założenia są realizowane.

Spotkanie z klientem w celu omówienia bieżącego projektu (**Wiktor**):

* Lider spotka się z klientem, aby omówić aktualny stan projektu i upewnić się, że potrzeby i oczekiwania klienta są spełniane.
* Lider przedstawi informacje na temat postępów w realizacji projektu i odpowie na wszelkie pytania i wątpliwości, jakie może mieć klient.
* Lider zbierze informacje zwrotne od klienta i dokona zmian w planach projektu.

Lider omawiający wynik spotkania z zespołem (**Wiktor, Daniel, Paweł, Michał, Sebastian**):

* Lider spotka się z zespołem, aby omówić wszelkie obszary, w których potrzebne są zmiany i poprawki na żądanie klienta.
* Lider będzie zabiegał o informacje zwrotne od reszty zespołu i rozważał ich sugestie dotyczące poprawek.
* Lider będzie współpracował z zespołem, aby wprowadzić wszelkie niezbędne poprawki do planu projektu w oparciu o otrzymane informacje zwrotne od nich i od klienta.

**Faza 5: Testowanie (30 dni, koszt: 20 000)**

Programiści wdrażają końcowe poprawki błędów i przeprowadzają dokładne testy (**Sebastian, Paweł**)

* Programiści przeprowadzają testy jednostkowe, aby zapewnić prawidłowe działanie poszczególnych komponentów aplikacji
* Programiści przeprowadzają kolejne testy, aby zweryfikować ogólną wydajność systemu i zidentyfikować podstawowe problemy.

Testowanie (outsourcing) (**Wiktor**):

* Lider będzie koordynował współpracę z zewnętrznym zespołem testującym w celu przeprowadzenia dodatkowych, dokładniejszych testów aplikacji.
* Zespół testujący wykona szereg testów, w tym testy funkcjonalne, wydajnościowe, bezpieczeństwa, kompatybilności i użyteczności.
* Lider przejrzy wyniki testów i będzie współpracował z programistami, aby naprawić wszelkie odkryte problemy lub błędy.

Poprawki **(Sebastian, Paweł, Michał**):

* Programiści i analityk będą pracować razem, aby naprawić wszelkie problemy lub błędy, które zostały odkryte podczas procesu testowania.
* Zespół upewni się, że wszystkie poprawki są prawidłowo przetestowane i poprawione przed przejściem do następnej fazy.

**Faza 6: Wdrożenie (1 dzień, koszt: 5 000)**

Aplikacja przekazana liderowi (**Wiktor, Sebastian**):

* Programiści przekażą gotową aplikację liderowi.
* Lider dokona przeglądu aplikacji, aby upewnić się, że spełnia ona wszystkie wymagania i funkcjonuje zgodnie z oczekiwaniami.

Spotkanie zespołu w celu omówienia ukończonej aplikacji (**Wiktor, Daniel, Michał**):

* Lider zwoła spotkanie z projektantem i analitykiem w celu omówienia ukończonej aplikacji.
* Dokonają oni przeglądu produktu końcowego i upewnią się, że spełnia on wszystkie wymagania i cele, które zostały ustalone.
* Zespół omówi wszelkie ostateczne szczegóły lub problemy, które należy rozwiązać przed wdrożeniem.

Poprawki (**Paweł, , Wiktor**):

* Programiści i lider dokonają ewentualnych poprawek w oparciu o informacje uzyskane podczas spotkania.
* Lider upewni się, że wszystkie poprawki zostały prawidłowo zaimplementowane.

Spotkanie z klientem w celu zaprezentowania aplikacji (**Wiktor, Daniel, Michał**):

* Lider, projektant oraz analityk spotkają się z klientem, aby zaprezentować ukończoną aplikację.
* Analityk zademonstruje klientowi cechy i funkcjonalność aplikacji.
* Projektant przedstawi finalną wersję wizualną aplikacji.
* Lider zbierze informacje zwrotne od klienta i odpowie na wszelkie pytania końcowe lub wątpliwości.

**Faza 7: Zakończenie (1 dzień, koszt: 2.000)**

Spotkanie zespołu w celu złożenia gratulacji liderowi (**Wiktor, Daniel, Michał, Paweł, Sebastian**):

* Lider zwoła spotkanie z zespołem, aby świętować pomyślne zakończenie projektu.
* Zespół pogratuluje liderowi udanego zarządzania projektem i podziękuje mu za jego ciężką pracę.
* Zespół zastanowi się również nad sukcesami i wyzwaniami projektu oraz omówi wszelkie wyciągnięte wnioski, które można zastosować w przyszłych projektach.

Końcowy przegląd projektu (**Wiktor**):

* Lider przeprowadzi końcowy przegląd projektu, aby upewnić się, że wszystkie cele i założenia zostały spełnione.
* Lider dokona przeglądu budżetu i harmonogramu, aby upewnić się, że projekt został ukończony na czas i w ramach przydzielonych zasobów.
* Lider udokumentuje wszelkie ustalenia końcowe lub zalecenia dotyczące przyszłych projektów.

Zamknięcie projektu (**Wiktor**):

* Lider zakończy wszelkie końcowe zadania administracyjne, w tym zamknie budżet projektu i uzupełni wszelką niezbędną dokumentację.
* Lider dziękuje zespołowi za jego ciężką pracę i wkład w realizację projektu.

**Podsumowanie**:  
Wydane 250 tysięcy   
Czas w jakim tego dokonano - 159 dni

**Wykorzystywane technologie/metodyki:**  
GitHub, GIT, Android studio, Postman, Java, datacomp, SCRUM/Agile, Microsoft Teams, Microsoft Office, Microsoft Excel, Microsoft Access, Adobe Creative Cloud, JIRA

**Bezpieczeństwo:**

* W celu zapewnienia jak największego bezpieczeństwa stosujemy ochronne baz danych DAM.
* Wszystkie połącznia telefoniczne są szyfrowane poprzez wysoko certyfikowane oprogramowanie VPN.
* Bezpieczeństwo komunikacji zapewnione przez mechanizmy szyfrowania (SSL/TLS) oraz wykorzystanie PKI (Public Key Infrastructure).

**Testowanie:**

* Testowanie funkcjonalności: sprawdzenie, czy wszystkie funkcje takie jak rezerwacja kuriera, śledzenie paczki i otrzymywanie powiadomień o dostawie działają prawidłowo.
* Testowanie kompatybilności: testowanie aplikacji na różnych urządzeniach i systemach operacyjnych, aby upewnić się, że jest ona kompatybilna z urządzeniami i platformami, które ma obsługiwać. (z założenia nie wymagane, jednak umożliwia to przyszły rozwój klientowi)
* Testy wydajności: testowanie wydajności aplikacji przy różnych obciążeniach i warunkach, takich jak duża liczba współbieżnych użytkowników, w celu zapewnienia, że może ona obsłużyć oczekiwane poziomy użytkowania.
* Testy bezpieczeństwa: testowanie środków bezpieczeństwa aplikacji, takich jak jej procesy uwierzytelniania i autoryzacji, aby zapewnić ochronę wrażliwych danych użytkowników.
* Testy użyteczności: testowanie doświadczenia użytkownika aplikacji w celu zapewnienia, że jest ona łatwa w użyciu i intuicyjna dla użytkowników.

**Protokoły członków zespołu oraz protokół finalny:**

**Protokół lidera (Wiktor)**  
**Temat: Lider omawiający wynik spotkania z zespołem**

Protokół przedstawia przegląd realizacji spotkania pomiędzy liderem, programistami, projektantem oraz analitykiem, podczas którego dokonano omówienia możliwych ulepszeń i poprawek w projekcie po spotkaniu z klientem.

Przed rozpoczęciem dyskusji lider przedstawił wszystkie informacje dotyczące projektu i jego postępów, oraz wyniku rozmowy z klientem. Następnie lider stworzył agendę spotkania z tematami dyskusji i obszarami usprawnień według zaleceń klienta. Głównym celem spotkania było uzyskanie informacji zwrotnej od programistów, projektanta oraz analityka na temat obszarów, w których można by było dokonać usprawnień lub też, w jaki sposób projekt mógłby zostać ulepszony, zgodnie z życzeniami klienta, następnie dostosować planu projektu według tych założeń. Podczas spotkania lider zadawał pytania, aby uzyskać informacje zwrotne od zespołu oraz zachęcał do dyskusji na temat możliwych popraw i usprawnień zgodnych z żądaniami klienta. Sugestie zespołu zostały szczegółowo omówione, w oparciu o nie wprowadzono poprawki do planu projektu. Lider upewnił się, że wszystkie cele spotkania zostały osiągnięte i że wszystkie niezbędne korekty zostały wprowadzone do planu projektu.

Finalnie spotkanie lidera z zespołem zakończyło się sukcesem w zakresie identyfikacji obszarów możliwych do poprawy i wprowadzenia niezbędnych korekt do planu projektu zgodnie z zaleceniami klienta.

**Protokół z tworzenia panelu logowania (Sebastian)**

Na etapie “tworzenie panelu logowania” spędziłem łącznie 7 dni.

Pierwsze dwa dni spędziłem nad wizualnym aspektem strony. Na początku skonsultowałem szczegółowe wymagania z naszym projektantem. Doszliśmy do porozumienia i wziąłem się za tworzenie strony wizualnej. Do strony logowania dodałem takie komponenty jak możliwość logowania za pomocą E- maila, możliwość zmiany hasła, szybkie odzyskiwanie hasła za pomocą E-maila, a także możliwość zarejestrowania się. Po konsultacji z projektantem dorzuciłem datę oraz czas w górnym prawym rogu ekranu a także logo naszej strony po przeciwnej stronie. Od strony wizualnej, pracowałem również nad animacją, która była zaplanowana na wcześniejszym etapie projektu.

3 dnia zająłem się stworzeniem formularza do rejestracji, dodałem do niego Komponenty pola tekstowego z możliwością dodania swoich danych oraz możliwość akceptacji regulaminów zawartych w naszej aplikacji. Po ukończeniu tego etapu rozpocząłem etap funkcjonalny.

Kolejne dni poświeciłem na napisanie funkcji poświęconych poszczególnym komponentom. Kolejno napisałem element kodu pobierający dane z formularza rejestracji oraz transportujący je zaszyfrowane do bazy danych, funkcję, która po zgłoszeniu chęci odzyskania hasła rozszyfrowuje zaszyfrowany element bazy danych i wysyła go na odpowiedni adres e-mail, Funkcję, która wymienia w bazie danych dane logowania.

**Protokół programisty (Paweł)**  
**Temat: Dodanie frontendu zgłaszania awarii/problemów**  
  
Protokół opisuje prace podjęte podczas tworzenia frontendu do systemu zgłaszania awarii/problemów. Praca nad tym rozwiązaniem trwała łącznie 7 dni.  
   
Na początku pracy przedstawiłem projektantowi makietę panelu zgłaszania awarii. Po naniesieniu poprawek, przeszedłem do programowania systemu zgłoszeniowego. Do systemu dodałem kolejno typ zgłoszenia, rodzaj awarii. W głównym panelu wyświetla się ilość wszystkich zgłoszeń, nowe zgłoszenia, ilość zgłoszeń, które są przypisane do pracownika oraz wykres tygodniowy nowych oraz zamkniętych zgłoszeń. Utworzono szablon informujący o przyjęciu zgłoszenia. Następnie zająłem się systemem zgłaszania awarii po stronie klienta. Utworzono formularz, w którym użytkownik wybiera rodzaj awarii, podaje dane kontaktowe, SLA zgłoszenia, oraz pole na opisanie awarii. Na końcu zająłem się zająłem się stworzeniem powiązania zgłoszenia do użytkownika z bazy danych. Zaprogramowano połączenie systemu zgłoszenia do bazy danych, co pozwoli automatycznie przydzielić osobę zgłaszającą do zgłoszenia.

**Protokół projektanta (Daniel)**

**Wygląd wizualny aplikacji**

Sporządzony przeze mnie projekt dotyczy finalnego wyglądu aplikacji, którą otrzyma klient.

Finalnie nad samym zaprojektowaniem wyglądu aplikacji spędziłem łącznie 2 dni, lecz tak naprawdę przez cały okres tworzenia aplikacji przez programistów były różne konsultacje i naniesione zmiany, które zostały zaakceptowane przez cały zespół. Cały opis wyglądu aplikacji został przeze mnie sporządzony na pierwszej stronie dokumentu. Moja praca polegała także na uczestniczeniu w spotkaniach z zespołem sprawdzając postępy oraz akceptując wraz z liderem dane etapy projektu, dodatkowo także uczestniczyłem w spotkaniach z klientem omawiając z nim obecny etap projektu, prezentując projekt oraz ustalanie dalszych założeń.

**Protokół finalny**

Protokół przedstawia przegląd planów oraz realizacji projektu, którym było zaprojektowanie oraz stworzenie aplikacji kurierskiej dla klienta poprawiające efektywność rozwożenia paczek przez kurierów. Zespół składał się z 2 programistów, projektanta, analityka oraz lidera zespołu. Projekt był prowadzony przy użyciu metodologii SCRUM/Agile w celu zapewnienia dostarczenie aplikacji w jak najwyższej jakości. W związku z tym odbywały się regularne spotkania całego zespołu w celu omawiania projektu oraz możliwości jego rozwoju. Naturalnie równie regularnie kontaktowano się z klientem przedstawiając mu obecne wyniki w rozwoju aplikacji w celu upewnienia się, iż klient jest zadowolony z przebiegu tworzenia aplikacji oraz spełnia ona jego założenia przez cały okres projektu.

Projekt został podzielony na fazy w celu optymalizacji efektywności planowania działań zespołu. W pierwszej fazie, projektant wraz z programistami oraz analitykiem skupił się na stworzeniu kilku projektów aplikacji oraz zaplanowali możliwe sposoby rozwoju funkcjonalności oraz interfejsu. Pod uwagę wzięto takie czynniki jak: zapotrzebowanie klienta, dostępne technologie oraz umiejętności zespołu. Podczas spotkania z klientem wybrano konkretny projekt. Lider po wstępnych spotkaniach stworzył zakres i cele oraz plan ich osiągnięcia w ramach określonego terminu oraz dostępnego budżetu, dokonał także zakupu niezbędnych licencji. Początkowa faza rozwoju obejmowała utworzenie frontendu aplikacji. Zaprojektowane strony logowania (podstawowa, linie papilarne oraz rozpoznawanie twarzy) zostały dodane, na tamten moment bez funkcjonalności backendowej. Deweloperzy utworzyli także frontendową część aplikacji do zgłaszania awarii/problemów oraz zgłaszanie problemów z paczką. Projektant przejrzał projekt, a następnie przekazał informacje programistą o potrzebnych poprawkach wizualnych, aby projekt był atrakcyjny wizualnie i przyjazny użytkownikowi w ramach obsługi aplikacji. Programiści przeprowadzili także kilka wstępnych testów, aby upewnić się, że nie ma żadnych poważniejszych błędów. Kolejną fazą była faza rozwoju aplikacji, podczas której programiści skupili się ona na dodaniu backendu zgłaszania awarii/problemów oraz backendu zgłaszania problemów z paczkami. Analityk ściśle współpracował z programistami, aby upewnić się, iż zaplanowane funkcje zostały prawidłowo zintegrowane z aplikacją. Tak jak w poprzedniej fazie odbywały się regularne spotkania zespołu oraz z klientem zgodnie z założeniami Agile/SCRUM, aby upewnić się, iż projekt jest na dobrej drodze, a założone cele i terminy są osiągane, a projekt spełnia wymagania klienta. W razie problemów, analityk, projektant oraz programiści są w stanie w kilka dni dokonać analizy problemu zgłoszonego przez klienta i dokonać poprawy aplikacji. W ramach integracji i usprawnienia aplikacji, programiści dokonali utworzenia baz danych, a także ich integracji oraz integracji aplikacji z google maps w celu utworzenia planów wyznaczania trasy dostarczania paczek. Dodali oni także funkcje backendową do systemu logowania się w każdy sposób, który został uwzględniony w projekcie (zwykłe logowanie, linie papilarne oraz rozpoznawanie twarzy). Tak jak w poprzedniej fazie analityk ściśle współpracował z programistami upewniając się, iż zaawansowane funkcje są prawidłowo zintegrowane z aplikacją. Dodatkowo projektant z współpracą z programistami skupił się na rozwoju wizualnym aplikacji, aby poprawić doświadczenie użytkowników i upewnić się, że jest ona atrakcyjna wizualnie i przyjazna dla użytkownika. Odbyły się także regularne spotkania, tak jak w poprzednich fazach rozwoju. Następnie w ramach rozwoju aplikacji niezbędnym elementem było przeprowadzenie testów. Składało się to z testów przeprowadzonych przez programistów z zespołu oraz zatrudnienie firmy zewnętrznej do przeprowadzenia dokładnych testów, aby upewnić się, iż aplikacja posiada minimalną ilość błędów. Po okresie testowania, programiści wprowadzą poprawki w aplikacji oraz przeprowadzą ich testy, aby upewni się, iż zostały poprawnie zaimplementowane. Po tym aplikacja została przekazana liderowi, który dokonał jej wstępnej analizy, aby upewnić się, że spełnia ona wszystkie wymagania i funkcjonuje zgodnie z oczekiwaniami. Odbyło się spotkanie całego zespołu, w ramach którego, dokonano finałowego przeglądu aplikacji, aby upewnić się, iż spełnia ona wszystkie funkcje i założenia, które zostały ustalone na początku oraz w trakcie jej rozwoju. Dokonano poprawek, które zostały ustalone podczas danego spotkania i upewniono się, iż działają one prawidło, a następnie aplikacja została przedstawiona klientowi przez lidera, projektanta oraz analityka, aby w pełni ukazać aspekty wizualne oraz możliwe funkcjonalności aplikacji, natomiast lider udzielił odpowiedzi na wszelkie możliwe pytania klienta. Na sam koniec odbyło się finalne spotkanie zespołu w ramach danego projektu, podczas którego lider uzyskał gratulacje za przeprowadzenie projektu, a także dokonano przeglądu budżetu i harmonogramu, aby upewnić się, że projekt został ukończony na czas i w ramach przydzielonych zasobów oraz udokumentowane wszelkie ustalenia końcowe lub zalecenia dotyczące przyszłych projektów a lider podziękował zespołowi za jego wkład w proces tworzenia aplikacji. Lider zakończy wszelkie końcowe zadania administracyjne, w tym zamknie budżet projektu i uzupełni wszelką niezbędną dokumentację.