1. **Struktura i system prac**

Ze względu na światową pandemię COVID-19 projekt był tworzony w niezwykle niesprzyjających warunkach, w obliczu powszechnie panujących obostrzeń, zwłaszcza w zakresie spotkań fizycznych, a więc również w swobodnej komunikacji i współpracy.

* 1. **Struktura prac**

Niniejszą pracę jak i projekt można podzielić na dwie ogólne części:

* Opis i analizy uwarunkowań społeczno-prawnych,
* Realizacja projektowo-techniczna.

Pierwsza z nich jest rozwinięta głównie na początku, są to rozdziały 1, 2, 4 i 9. Natomiast druga część to rozdziały 3 i 5-9.

Podział wynika z naturalnych potrzeb pozyskania informacji, zasięgnięcia opinii, jak również zastanowienia się nad własnymi koncepcjami i wizjami realizacyjnymi wynikającymi z refleksji, analizując dane zagadnienie. Stąd też zagadnienia poruszane na początku mają ogromny wpływ na drugą część, gdzie należy zaprojektować system informatyczny oraz dobrać odpowiednie narzędzia tak, aby sprostać wymaganiom postawionym na początku. Niekiedy okazywało się, że już w trakcie prac implementacyjnych należało zmienić lub przebudować niektóre z pierwotnych koncepcji, a co za tym idzie strukturę pracy.

* 1. **System prac**

Całość prac rozpoczęła się w lipcu 2021 r. i można podzielić ją na 4 etapy:

* Pierwszym etapem pracy były wspólne dyskusje i wzajemne wizje realizacji projektu, aby ostatecznie wyłonić jedną wspólną koncepcję;
* Następnie należało poszerzyć wiedzę z danej dziedziny oraz zebrać opinie, co każdy z członków robił w ramach swoich możliwości indywidualnie, a następnie dzielił się pozyskaną wiedzą i informacjami;
* Najobszerniejszym i zarazem najtrudniejszym elementem była faza implementacji. Należało wybrać odpowiednią technologię oraz zapoznać się z nią, uzyskując znaczny stopień biegłości. Był to zdecydowanie najdłuższy etap, zakładał między innymi czas na dokształcanie się w ramach używanych technologii. Podczas realizacji powstawało wiele błędów i problemów natury technicznej, których rozwiązanie często obarczone było dużymi nakładami czasu i energii, przyczyniając się jednak do poszerzenia wiedzy;
* Etap testowania, dopracowywania oraz poprawa wykrytych błędów. Każda z osób odpowiedzialna była za inne sposoby i typy testów.

Do realizacji projektu (części tworzenia oprogramowania) zaadoptowanych zostało wiele idei zwinnej metodyki Scrum. Według tej techniki określono w zespole Mistrza Scrum (ang. *Scrum Master*), Właściciela Produktu (ang. *Product Owner*) i zespół odpowiedzialny za rozwój oprogramowania (ang. *Software Development Team*):

* Właściciel Produktu - Wojciech Lewandowski
* Mistrz Scrum - Katarzyna Kulesa
* Zespół Rozwoju - Wojciech Koszela oraz Mateusz Kluba

Najważniejszą cechą zwinnych metod zarządzania projektami to ich elastyczność, a co za tym idzie wydajność. W tym przypadku należało zwrócić uwagę na pewne aspekty, których w tym projekcie nie było, jak chociażby brak realizacji produktu pod konkretnego klienta, co wiązałoby się już z pewnymi odgórnie sprecyzowanymi wymaganiami. W omawianej sytuacji rola zespołu była trudniejsza, ponieważ musiał samodzielnie określić wszelkie wymagania, postawić się na miejscu klienta oraz zestawić je z oczekiwaniami i rzeczywistością wdrożeń. Z uwagi na skalę prac należało (zwłaszcza na początku) zaangażować wszystkie osoby do tworzenia produktu, jak również do pozyskiwania informacji i budowania spójnej koncepcji. Dlatego też w pewnych aspektach nie był to ścisły podział, a każdy z członków starał się wspomóc innych w ich obecnie realizowanym celu wedle swojej najlepszej wiedzy i umiejętności. Dawało to bardzo dobre efekty w postaci widocznych postępów prac.

Roboczo wyodrębnione zostały również dwa podzespoły: odpowiedzialny za frontend (Katarzyna Kulesa i Mateusz Kluba) oraz backend (Wojciech Koszela i Wojciech Lewandowski). Ponadto każdy z podzespołów miał osobę odpowiedzialną za ostateczny kształt określonego na dany tydzień zadania przed spotkaniem całego zespołu. Na płaszczyźnie frontendowej była to Katarzyna Kulesa, a backendowej Wojciech Koszela.

Prace prowadzone były w ramach tygodniowych sprintów (iteracji), zawierających konkretne zadania wyznaczone dla członków zespołu. Każdy ze sprintów podsumowany był na cotygodniowych zebraniach w ramach całego zespołu, a następnie wyznaczane zostawały kolejne zagadnienia do rozwiązania (lub poprawki do zadań obecnych). Za wyznaczanie i koordynację spotkań odpowiedzialny Właściciel Produktu przy współpracy z Mistrzem Scrum.

Komunikacja podczas poszczególnych iteracji mocno nastawiona była na pracę w parach (podzespołach). Ułatwiało to komunikację, przejrzyście określało zakres prac, w tym także poszerzanie wiedzy z zakresu odpowiedniej technologii. Skutkowało to wyspecjalizowaniem się każdego z członków zespołu, prowadząc do szybszej realizacji kolejnych zadań i sprawniejszego rozwiązywania błędów. Takie podejście pozwoliło jeszcze bardziej usprawnić pracę, gdyż wiele zadań można było realizować niezależnie i dopiero na wspólnym spotkaniu omawiać ich integrację.

Komunikacja grupowa prowadzona była praktycznie w całości za pośrednictwem narzędzi do komunikacji zdalnej, głównie aplikacji *Discord*. Umożliwia on spotkania całego zespołu na prywatnym kanale głosowym, dającym również możliwość wspólnego czatu. Dzielenie się wiedzą ułatwiała możliwość współdzielenia ekranu, co jest niezwykle istotne podczas omawiania problemów (zwłaszcza deweloperskich) i wspólnym szukaniu jego rozwiązań.

Do określenia konkretnych zadań w ramach sprintu wykorzystywany był serwis *Trello*, który umożliwia podpięcie każdemu z członków zespołu konkretnego zadania, wraz z jego opisem, wymaganiami i terminem realizacji. Pozwala również śledzić bieżące postępy, gdy osoba je realizująca informuje o tym poprzez przenoszenie zadania do kolejnych zakładek symbolizujących jego stan gotowości.

Całość prac deweloperskich realizowana była za pomocą systemu kontroli wersji *Git*, tworząc prywatne i dedykowane repozytorium na *GitHubie*. Każde zadanie (ang. *task*) realizowane było w ramach osobnej gałęzi (ang. *branch*). Umożliwiło to wszystkim członkom jednoczesną pracę nad własnymi, kolejnymi problemami, bez ryzyka ingerencji i negatywnych skutków w gotową główną gałąź projektu (ang. *master*). Do łączenia (ang. *merge*) gałęzi, po wcześniejszym wykonaniu i zatwierdzeniu danego zadania, upoważniony był tylko Mistrz Scrum lub osoba, która taką czynność skonsultowała z resztą zespołu. Odpowiadał on też za akceptację i nadzór nad spójnym i logicznie powiązanym rozdzielaniem zadań. Za koordynację konceptualnych wizji i planów projektu odpowiedzialny był Właściciel Produktu.

Technologią wybraną do realizacji projektu został framework Django (Python), w którym napisany został kod po stronie backendu, za frontend odpowiada Vue.js. Lokalne prace odbywały się w środowisku programistycznym Visual Studio Code na systemie operacyjnym Windows 10 lub Linux.

W tworzenie samej pracy (pisemnej) zaangażowany był każdy z członków zespołu z uwzględnieniem głównych zadań, jakimi zajmował się podczas tworzenia produktu:

* Katarzyna Kulesa:
  + 1. Wprowadzenie
  + 1.1. Wstęp
  + 2. Motywacja i cele
  + 2.1. Motywacje
  + 3. Struktura i system prac
  + 3.2. System prac
  + 6. Architektura systemu
  + 6.2. Komunikacja frontendu i backendu
  + 8. Użytkowanie systemu
  + 8.1 Wygląd i działanie
  + 9. Podsumowanie
  + 9.1 Wnioski
* Wojciech Lewandowski:
  + 1. Wprowadzenie
  + 1.2. Definicja konsultacji
  + 1.3. Perspektywy
  + 4. Analiza
  + 4.1. Podstawowe zagadnienia ogólne
  + 5. Wymagania i działanie systemu
  + 5.3. Diagram przypadków użycia
  + 7. Kod i oprogramowanie
  + 7.2. Konwencja pisania kodu i struktura plików
  + 7.3. Bezpieczeństwo
* Wojciech Koszela:
  + 2. Motywacja i cele
  + 2.2. Cele
  + 3. Struktura i system prac
  + 3.1. Struktura prac
  + 5. Wymagania i działanie systemu
  + 5.1. Aktorzy
  + 7. Kod i oprogramowanie
  + 7.1. Zaimplementowane rodzaje głosowań
  + 7.4. Testowanie
  + 7.5. Aspekt wymogów technicznych
* Mateusz Kluba:
  + 4. Analiza
  + 4.2. Rozmowy konsultacyjne
  + 4.3. Spostrzeżenia i refleksje
  + 5. Wymagania i działanie systemu
  + 5.2. Diagram kontekstu
  + 6. Architektura systemu
  + 6.1. Schemat bazy danych
  + 9. Podsumowanie
  + 9.2. Dalszy potencjał rozwojowy i plany
  + 10. Bibliografia