

Raport retrospekcyjny z HackatIOnu

Dziennik pokładowy projektu Student Traffic

Krzysiu i Spółka

11–13 kwietnia 2025

Dzień 1 – 11.04.2025

11:30–12:00 – Wprowadzenie

Wysłuchaliśmy zasad zaliczenia oraz spraw organizacyjnych związanych z projektem. Ustalono ramy czasowe – 2 dni intensywnej pracy.

12:00–13:30 – Wybór lidera, burza mózgów i wybór pomysłu

Zespół zdecydował również o wyłonieniu lidera projektu – został nim Krzysztof, który podjął się koordynowania działań oraz kontaktu z klientem. W sześciuosobowym zespole przeprowadziliśmy burzę mózgów. Pojawiły się pomysły takie jak:

- Aplikacja do szukania współlokatora (na wzór Tindera),
- Aplikacja do śledzenia pozycji Jaguarów na miasteczku AGH (inspirowana Yanosikiem),
- Aplikacja typu All-in-One dla mieszkańców akademików.

Zdecydowaliśmy się na ostatnią opcję, ponieważ najlepiej wpisywała się w realne potrzeby studentów.

Student Traffic to aplikacja mobilna, która powstała z myślą o studentach mieszkających w akademikach Miasteczka Studenckiego AGH. Rozwiązuje ona szereg problemów codziennego życia na kampusie – od braku wolnych pralek, przez utrudnioną komunikację, aż po organizację wydarzeń.

Główne cele aplikacji:

- Usprawnienie codziennych czynności (np. rezerwacja pralek, suszarek, boisk),
- Budowanie lokalnej społeczności studenckiej (moduł ogłoszeń i wydarzeń),
- Promowanie aktywności i integracji (możliwość tworzenia wydarzeń sportowych),
- Zwiększenie dostępności informacji (wszystko w jednej aplikacji).

13:30–14:30 – Podział prac i wybór technologii

Po zatwierdzeniu pomysłu przez klienta rozpisaliśmy kluczowe funkcjonalności i ustaliliśmy ich priorytety. Nie wprowadzaliśmy sztywnego podziału na osoby zajmujące się frontendem i backendem. Zespół dynamicznie dostosowywał się do potrzeb, a członkowie przejmowali zadania tam, gdzie było najwięcej pracy lub potrzebna była pomoc.

Wybrane technologie:

- Frontend: React + shadcn/ui,
- Backend: Spring Boot,
- Baza danych: Firebase Realtime Database.

Dzień 2 – 12.04.2025

09:00–11:00 – Start prac technicznych

Z samego rana przystąpiliśmy do właściwej implementacji aplikacji.

Frontend: rozpoczęto pracę nad stroną główną aplikacji, a także przygotowano makiety pozostałych ekranów, co pomogło ustalić spójny wygląd i UX aplikacji. Zespół frontendowy pracował w ścisłej komunikacji z backendem, aby dobrze zrozumieć strukturę danych.

Backend: równolegle rozpoczęto projektowanie struktury bazy danych oraz testowano możliwość integracji Spring Boota z Firebase, co okazało się bardziej problematyczne niż pierwotnie zakładano.

11:00–13:00 – Tworzenie komponentów

Frontend: stworzono pierwsze interaktywne komponenty – podstrony do rezerwacji pralek i suszarek oraz wstępną wersję widoku wydarzeń. Szczególną uwagę zwrócono na prostotę interfejsu i intuicyjność użytkowania.

Backend: rozpoczęto implementację metod do komunikacji z bazą danych – z uwagi na brak oficjalnego wsparcia dla Firebase w Springu, część integracji wymagała napisania własnych rozwiązań i dostosowania zapytań do niestandardowego formatu bazy.

13:00–13:30 – Przerwa obiadowa

Czas na odpoczynek i zebranie sił na dalsze prace.

13:30–15:30 – Dalsza implementacja

Frontend: kontynuowano implementację kolejnych ekranów aplikacji, w tym widoków ogłoszeń i listy wydarzeń. Prace posuwały się wolniej ze względu na ograniczone doświadczenie niektórych członków zespołu, jednak dzięki wzajemnemu wsparciu oraz pomocy bardziej doświadczonych osób udało się utrzymać tempo.

Backend: zespół napotkał problemy ze stabilną i wydajną integracją z Firebase. Wymagało to wielu prób i testów różnych podejść, by móc uzyskać komunikację spełniającą wymagania aplikacji.

15:30–16:30 – Logowanie i integracja

Frontend: rozpoczęto implementację ekranu logowania, który miał być kluczowym punktem wejściowym do aplikacji. Wprowadzono formularz oraz interfejs do obsługi sesji użytkownika.

Backend: jedna z osób z zespołu frontendowego, która miała większe doświadczenie z Javą, dołączyła do zespołu backendowego. Udało się stworzyć działające endpointy do logowania i obsługi rezerwacji pralek, co umożliwiło rozpoczęcie faktycznej integracji między frontem a backendem.

16:30–17:00 – Podsumowanie dnia i code-review

Na koniec dnia odbyło się wspólne spotkanie zespołu, podczas którego każdy członek zespołu podzielił się postępami oraz napotkanymi problemami. Przeprowadzono wzajemny przegląd kodu (code-review), co pozwoliło poprawić jakość projektu i zadbać o spójność stylu kodowania. Wspólnie podjęto decyzje dotyczące dalszego rozwoju i rozdziału zadań na ostatni dzień hackathonu.

Dzień 3 – 13.04.2025

09:00–11:00 – Nowe moduły i poprawki

Frontend: rozpoczęto dzień od implementacji ekranu ogłoszeń. Dodatkowo wykonano refaktoryzację stylów i komponentów do spójnych ze standardem shadcn/ui, co poprawiło wygląd i strukturę całej aplikacji.

Backend: po analizie wcześniejszej struktury danych zdecydowano się na jej przemodelowanie, aby lepiej odpowiadała rzeczywistym wymaganiom. Ukończono również logikę związaną z rezerwacjami pralek i suszarek.

11:00–13:00 – Rozszerzenia i łączenie

Frontend: udało się skutecznie połączyć frontend z backendem w zakresie rezerwacji, co pozwoliło na pełne przetestowanie tej funkcji. Rozbudowano także widok wydarzeń – użytkownik może teraz wybierać rodzaj aktywności (np. piłka nożna, siatkówka).

Backend: kontynuowano prace nad nowymi modułami – dodano obsługę wydarzeń, ogłoszeń oraz możliwość zgłaszania awarii. Dzięki zebranych informacjom z frontendu dostosowano odpowiednio strukturę danych i endpointy.

13:00–13:30 – Przerwa obiadowa

Ostatnia wspólna przerwa — czas na zebranie sił przed finałowym sprintem.

13:30–16:30 – Finalizacja i testy

Frontend: zespół frontendowy zintegrował wszystkie dotychczas stworzone moduły w spójną aplikację. Dodano ekran zgłaszania awarii oraz podgląd zarezerwowanych wydarzeń. Przepro-

wadzono testy manualne, poprawiono drobne błędy i zadbano o szczegóły stylistyczne.

Backend: zakończono implementację logiki wydarzeń i dopracowano komunikację z frontendem. Jedna osoba z backendu wsparła także integrację frontendową, co przyspieszyło ostatni etap przygotowań.

16:30–17:30 – Prezentacja projektu

Zespół zaprezentował działający prototyp aplikacji Student Traffic. Projekt zawierał wszystkie planowane funkcjonalności, takie jak: logowanie, rezerwacje pralek i boisk, ogłoszenia, wydarzenia oraz system zgłaszania usterek. Jak to na hackathonie – nie wszystko działało idealnie, a niektóre elementy mogłyby być bardziej dopracowane lub rozwinięte. Jednak w ciągu zaledwie 20 godzin intensywnej pracy udało się stworzyć kompletny prototyp gotowy do demonstracji.

Podsumowanie

Hackathon był intensywnym doświadczeniem, które wymagało dobrej organizacji pracy zespołowej, elastyczności i szybkiego uczenia się. Udało się osiągnąć zamierzony cel i stworzyć działającą aplikację spełniającą rzeczywiste potrzeby studentów AGH.

Sz szczególnie cenne było doświadczenie pracy w dynamicznie zmieniających się warunkach, elastyczne podejście do podziału zadań oraz wzajemna pomoc i otwarta komunikacja w zespole. Mimo ograniczonego czasu i napotkanych problemów technicznych, zespół poradził sobie z każdym wyzwaniem, co przełożyło się na satysfakcjonujący efekt końcowy.