

Załącznik nr 1 – Zakres merytoryczny „Kurs Python + Django + AI – weekendowy”

Informacje ogólne dotyczące szkolenia

Celem szkolenia jest przygotowanie osób do podjęcia pracy jako Junior Python Developer. Ma ono wyposażyć uczestników w znajomość technologii, języków i narzędzi, niezbędnych do pracy, ale także wyrobić dobre nawyki i solidny warsztat programistyczny.

W czasie całego cyklu nauczania staramy się dać adeptom nie tylko wiedzę, ale przede wszystkim możliwość zrozumienia, jak wygląda codzienna praca dobrego programisty. Wszystkie zajęcia mają więc formę warsztatową i prowadzone są przez doświadczonych praktyków, specjalistów w wybranych dziedzinach.

Zajęcia trwać będą przez 20 weekendów. Każdy weekend w Akademii będzie miał podobną formę. W sobotę odbywać się będą zajęcia warsztatowe, na których uczestnicy nabywać będą nowe umiejętności. W niedzielę dzień podzielony będzie na 3 godziny warsztatowe na temat narzędzi bądź dobrych praktyk oraz 3 godziny projektowe. W ramach projektów uczestnicy będą mieli możliwość wykorzystania zdobytych kompetencji w pracy zespołowej.

Wymagania sprzętowe: system operacyjny Linux lub macOS.

Optymalna konfiguracja sprzętowa: procesor i5+ lub podobny, 8GB+ pamięci RAM, zalecany dysk SSD.

Moduły szkoleniowe

Poniżej znajduje się lista modułów szkoleniowych, które zostaną zrealizowane podczas kursu „Junior Python Developer” w Akademii infoShare.

W przypadku każdej z grup szkoleniowych program będzie dostosowywany do poziomu zaawansowania grupy, także w trakcie trwania szkolenia.

1. Narzędzia programistyczne

Poznanie podstawowych narzędzi programistycznych, używanych w codziennej pracy programisty, ze szczególnym uwzględnieniem IDE (PyCharm), podstaw używania repozytorium kodu git (GitHub) i konteneryzacji (Docker).



0. Scrum

Reguły zwinnego wytwarzania oprogramowania, ze szczególnym uwzględnieniem metodyki Scrum.

0. Podstawy programowania

Wprowadzenie do programowania, na przykładzie języka Python. Podstawowe struktury danych oraz ich sprawne przetwarzanie.

0. Django

Wprowadzenie do tworzenia aplikacji webowych w oparciu o framework Django. Zapoznanie z architekturą aplikacji webowych oraz przedstawienie idei frameworka webowego.

0. Testowanie automatyczne

Automatycznie testowanie aplikacji na poziomie jednostkowym i integracyjnym. Idea i praktyczne zastosowanie Test Driven Development, w tym mockowanie obiektów. Nabycie umiejętności debugowania i profilowania kodu aplikacji.

0. Loggery

Praktyczne zastosowanie najpopularniejszych loggerów. Umiejętności wartościowego logowania istotnych informacji.

0. Refaktoryzacja

Poznanie idei, celu i zrozumienie potrzeby refaktoryzacji kodu oraz nabycie umiejętności jego realizacji przy aktywnym wsparciu przez IDE.

0. System kontroli wersji git

Warsztaty ze skutecznego używania systemu kontroli wersji git, w tym rozwiązywanie konfliktów, git flow oraz narzędzia wspomagające.

0. Przeglądy kodu

Zrozumienie idei, wartości i sztuki przeglądów kodu. Zrozumienie znaczenia czystości kodu oraz dbania o jego utrzymywalność.

0. Środowisko ciągłej integracji

Poznanie wartości środowiska ciągłej integracji oraz umiejętności efektywnego jego wykorzystania, na przykładzie systemu Jenkins.



0. UML i wzorce projektowe

Poznanie i zrozumienie znaczenia podstawowych diagramów UML w celu skuteczniejszej komunikacji z innymi developerami. Poznanie i zrozumienie podstawowych wzorców projektowych.

0. Budowanie i używanie REST API

Poznanie REST oraz nabycie umiejętności prawidłowego definiowania API oraz jego implementowania w języku Python.

0. Bazy danych

Wykorzystanie baz danych SQL do przechowywania danych.

0. Flask

Podstawy frameworku do tworzenia prostych aplikacji webowych.

0. AWS

Podstawy pracy ze środowiskiem chmurowych Amazonu.

