

Lekcja:Enter! - scenariusz szkolenia dla nauczycieli informatyki szkół ponadpodstawowych

WPROWADZENIE	2
MODUŁ 1.	5
Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych	5
1.1. Eclipse, czyli projekt z klasą	7
1.2. Budujemy bazę	12
MODUŁ 2.	17
Rozumienie, analizowanie i rozwiązywanie problemów	17
2.1 Algorytmy na Javie	18
2.2 Mobilne szyfrowanie	21
MODUŁ 3.	25
Zajęcia w szkole z własnymi scenariuszami	25
MODUŁ 4.	29
Omawiamy przeprowadzone zajęcia	29

WPROWADZENIE

Zgodnie z koncepcją projektu Lekcja:Enter celem szkolenia dla nauczycieli informatyki, które będzie prowadzone z pomocą niniejszego scenariusza, jest podniesienie kompetencji z zakresu nauczania kodowania, algorytmiki i programowania oraz pokazanie i przećwiczenie rozwiązań metodycznych do zastosowania w szkole. Uczestniczący w szkoleniach nauczyciele informatyki samodzielnie zaplanują dwie lekcje dotyczące dwóch pierwszych celów podstawy programowej informatyki (po jednym scenariuszu do każdego celu), skonsultują przygotowane scenariusze z trenerami lokalnymi, a po ich akceptacji – przeprowadzą według nich lekcje w szkole.

Całość procesu doskonalenia umiejętności nauczycieli została zaplanowana zgodnie z modelem wspomaganego uczenia się. Zaplanowane szkolenie odpowiada potrzebom nauczycieli w zakresie rozwijania kompetencji z zakresu programowania w procesie uczenia się uczniów. Zostało ono zbudowane z dwóch głównych elementów. Pierwszy element szkoleniowy obejmuje szkolenie stacjonarne oraz przygotowane zasoby edukacyjne dla nauczycieli służące podnoszeniu kompetencji w zakresie programowania, oraz element wdrożeniowy polegający na opracowaniu co najmniej dwóch scenariuszy zajęć dla uczniów z wykorzystaniem wiedzy i umiejętności poznanych na szkoleniu, w tym jeden z wykorzystaniem zasobów i platformy epodreczniki.pl.

Scenariusz szkolenia nauczycieli informatyki zakłada podstawowy poziom ich kompetencji programistycznych.

Szkolenie powinno być realizowane podczas pięciu dni szkoleniowych w układzie 2 x 2dni+1 dzień. Przed ostatnim dniem szkoleniowym, uczestnicy powinni przeprowadzić lekcje według wcześniej napisanych scenariuszy, stąd proponuje się większy odstęp czasowy pomiędzy spotkaniami: minimum 4 tygodnie, a maksimum 8 tygodni. Szkolenie można też przeprowadzić w układzie 3 dni +1 dzień, jednak trzeba mieć na uwadze, że te 3 dni będą czasem bardzo intensywnej pracy (około 9 godzin lekcyjnych/dzień).

Szkolenie będzie prowadzone w formie opartej o aktywny udział uczestników z wykorzystaniem sprzętu elektronicznego z dostępem do Internetu. Prowadzący będzie przechodził przez kolejne punkty szkolenia razem z uczestnikami, wspólnie wykonując poszczególne zadania. Każdy uczestnik szkolenia będzie wykonywał zadania związane z poszczególnymi etapami w obecności prowadzącego pod jego opieką, co podniesie jakość szkolenia i skuteczność nabywania kompetencji. Po każdym opracowanym zagadnieniu uczestnicy będą mogli zadawać prowadzącemu pytania dotyczące treści szkolenia.

Ważne, aby osoba prowadząca przed szkoleniem zapoznała się ze scenariuszem oraz sprawdziła, jakie aplikacje, programy będą wykorzystywane podczas szkolenia.

Obowiązkiem trenera/trenerki jest przygotowanie wszystkich komputerów w miejscu szkoleniowym do pracy. Jeśli uczestnicy szkolenia będą chcieli pracować na własnych laptopach, to należy ich wcześniej poinformować jakie oprogramowanie i w jakiej wersji należy zainstalować.

Uzupełnieniem szkolenia będą następujące materiały dla nauczycieli:

1) Pomocnik szkoleniowy, pokazujący jak krok po kroku wykonać ćwiczenia ze szkoleń

2) Cztery dłuższe tutoriale:

- Wprowadzenie do używania środowiska Jupyter,
- Podstawowe zagadnienia związane z programowaniem na przykładzie języka Python
- Omówienie (opis, instalacja, prosty przykład użycia) wybranych środowisk informatycznych oraz zasobów omawianych podczas szkolenia (Eclipse, Netbeans, MsAccess, Android Studio),
- Bazy danych - wstęp teoretyczny i konceptualny

3) 10 krótkich tutoriali wprowadzających do narzędzia Tinkercad Circuit, programowania blokowego (wizualnego) i tekstowego wirtualnej płytki Arduino

Z uwagi na funkcje wykonywane w szkole warto, żeby nauczyciele informatyki zainteresowali się procesem wdrażania planu TIK do działań dydaktycznych w szkole. Ich wiedza i umiejętności mogą być wsparciem zarówno dla dyrektora jak i innych nauczycieli biorących udział w projekcie. Nauczyciele informatyki mają doświadczenie w prowadzeniu zajęć z wykorzystaniem komputerów, potrafią przewidzieć trudne sytuacje jakie mogą się z tym wiązać, a w razie wystąpienia problemu, wiedzą jak go rozwiązać. Świadomość dyrektora i nauczycieli, że mają wsparcie specjalisty/specjalistki w dziedzinie TIK, pomoże przełamać strach przed korzystaniem z nowych technologii. Nauczyciel/nauczycielka TIK może też być znakomitym tutorem w procesie wzajemnego uczenia się nauczycieli.

W ramach projektu Lekcja:Enter wsparcie otrzymają także dyrektorzy szkół, z których nauczyciele uczestniczą w szkoleniach. W module online dla dyrektorów znajdzie się przewodnik i narzędziownik pt. „Aktywna szkoła z TIK” zawierający między innymi plan wdrażania TIK do działań dydaktycznych w szkole, omówienie sposobów jego wdrożenia, wskazanie roli dyrektora/dyrektorki w realizacji planu, a także opis działań promujących przedsięwzięcia szkoły w zakresie wdrażania TIK. W narzędziowniku znajdą się m.in. schemat planu, fragment przykładowego planu, przykładowe narzędzia ułatwiające monitorowanie wdrażania TIK w szkole oraz przykładowy fragment statutu szkoły umożliwiający korzystanie przez uczniów ze smartfonów w celach edukacyjnych w formule BYOD (*bring your own device*). Wsparciem dla dyrektorów będzie także cykl 4 webinarów, z których ostatnie będzie poświęcone doświadczeniom z przygotowania i realizacji planów w ich szkołach.

MODUŁ 1.

Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych

(1) Informacje organizacyjne dla trenerów

Grupa docelowa uczestników: nauczyciele informatyki w szkołach ponadpodstawowych

Liczebność grupy szkoleniowej: 8-12 osób

Poziom zaawansowania grupy docelowej: podstawowy

Moduł 1.szkolenia składa się z dwóch części:

- 1.1. Eclipse, czyli projekt z klasą (część online – 180 minut (4 x 45 minut); część stacjonarna – 360 minut (8 x 45 minut))
- 1.2, Budujemy bazę (330 minut – 7,8 x 45 minut)

(2) Moduł 1 – opis

Pierwszy moduł jest wejściem w tematykę szkolenia/projektu, ale zarazem pierwszym spotkaniem uczestników. W związku z tym trzeba szczególnie mieć na uwadze nie tylko treści merytoryczne, ale również względy procesowe. Konieczne jest zadbanie o to, aby osoby w grupie dobrze (bezpiecznie) się poczuły, zrozumiały, że czeka je długi wspólny proces, w czasie którego będą nie tylko siedzieć obok siebie, ale też korzystać nawzajem ze swojej wiedzy, doświadczeń, przemyśleń. Pierwsza sesja nie zapewni pełnej otwartości i zaufania, to wymaga czasu, ale może dać podwaliny pod tworzenie się grupy.

Dlatego też tempo pracy „merytorycznej” w tym module jest wolniejsze niż w kolejnych częściach szkolenia – po to, aby osoby uczestniczące w szkoleniu miały nieco więcej czasu na bycie razem. Oczywiście – cały czas w temacie programowania.

Część wstępna poświęcona jest poznaniu się uczestników, ale w kontekście tematu, którego dotyczy szkolenie. Ważnym momentem jest zebranie oczekiwań i obaw, gdyż każda osoba przychodzi z własną wizją tego, co będzie się działo i czego będzie dotyczyło szkolenie. To jest moment na wyjaśnienie i ewentualne

rozładowanie napięcia, aby wykluczyć sytuację, w której niektórzy uczestnicy czekali na realizację swoich celów i nie mogli się doczekać.

Ważnym elementem części wstępnej jest tworzenie kontraktu/zasad pracy – to jeden z tych momentów, gdy uczestnicy mogą powiedzieć o swoich potrzebach i stworzyć ramy do swojego funkcjonowania w grupie. To też narzędzie budujące grupę i relacje.

Dyskusja wokół zadanych pytań może mieć różny poziom szczegółowości i eksperckości – nie przejmuj się. Celem jest osadzenie uczestników w temacie i nakierowanie ich myślenia na korzyści płynące z nauki programowania oraz ich osobiste działania edukacyjne. Ta aktywność to również okazja do poznania/przypomnienia sobie różnych zasobów, z których uczestnicy korzystają ucząc algorytmiki i programowania.

Temu samemu służyć też ma flipczart (a docelowo może flipczarty), na których będą zapisywane te programy, aplikacje, serwisy, o których uczestnicy wspominają, a nie ma ich omówionych w programie szkoleniowym.

Dalsza część szkolenia będzie przebiegała wokół wspólnego uczenia się. Uczestnicy stworzą prostą aplikację konsolową w środowisku Eclipse oraz spróbują rozwiązać problem biznesowy w MS Access.

Ważnym elementem tej części jest zwrócenie uwagi na to, gdzie i jak można się uczyć od innych nauczycieli i nauczycielek (*peer-to-peer learning*) oraz podkreślenie korzyści płynących z takiego sposobu uczenia się.

Uczestnicy w sposób praktyczny, poprzez realizację wybranych projektów, poznają zasady programowania w obu środowiskach.

W części podsumowującej, należy zwrócić uwagę uczestników, że każdy moduł szkolenia, kończy się zadaniem wdrożeniowym. Ma ono różny charakter, może to być proste polecenie związane z aplikacją, analiza dobrych praktyk lub obejrzenie tutorialu w celu przygotowania się do następnych zajęć (strategia nauczania wyprzedzającego)

(3) **Cele modułu**

- Zapoznanie uczestników ze środowiskiem szkolenia: osobą prowadzącą, tematyką, wykorzystywanymi narzędziami
- Zapoznanie uczestników z platformą projektu, platformą ePodreczniki oraz programem szkolenia wraz z wykazem kompetencji do osiągnięcia przez uczestników,
- Zapoznanie uczestników z wybranymi środowiskami informatycznymi wspierającymi tworzenie różnego rodzaju aplikacji (Eclipse, Netbeans, MsAccess, Android Studio)

- Omówienie podstawowych zagadnień związanych z programowaniem na przykładzie języka Java
- Przedstawienie procesu projektowania i implementacji bazy danych przy użyciu oprogramowania MS Access
- Zapoznanie uczestników z możliwościami użycia wybranych technologii do realizacji zadań przewidzianych przez podstawę programową

(4) **Przebieg szkolenia**

1.1. Eclipse, czyli projekt z klasą

CZĘŚĆ ONLINE: (180 minut)

Przed rozpoczęciem szkolenia stacjonarnego, trener/trenerka prowadzący/prowadząca wysyła za pomocą platformy projektu informację do uczestników, aby zapoznali się z przygotowanym dla nich materiałem w ramach kształcenia wyprzedzającego.
(Wzór informacji zostanie opracowany podczas szkolenia dla trenerów lokalnych)

Część wstępna (online) - czas trwania: **30 minut**

1. Wyślij do uczestników wiadomość powitalną, przedstaw się, zamieść plan szkolenia, podaj uczestnikom link do platformy i poproś, aby się zalogowali. Wyjaśnij zalety kształcenia wyprzedzającego i zaproś uczestników do pracy z materiałem szkoleniowym dostępnym na platformie. Bądź w kontakcie z uczestnikami i odpowiadaj na ich pytania.

Część warsztatowa (online) - czas trwania: **150 minut**

4. Omówienie (opis, instalacja, prosty przykład użycia) wybranych środowisk informatycznych oraz zasobów używanych w dalszej części kursu (Eclipse, Netbeans, MsAccess, Android Studio) - czas trwania: **75 minut**
5. Bazy danych - wstęp teoretyczny i konceptualny - czas trwania: **30 minut**
 - Podstawowe pojęcia - baza danych, tabela, klucz podstawowy, klucz obcy, rekord, pole, relacja
 - Typy relacji - jeden do jednego, jeden do wielu, wiele do wielu
6. Platforma epodreczniki - utworzenie konta na platformie, zapoznanie się z dostępnymi zasobami dotyczącymi informatyki, utworzenie przykładowych materiałów w edytorze materiałów – czas trwania: **45 minut**

CZĘŚĆ STACJONARNA: (720 minut)

Część wstępna (ok. 45 minut)

1. Zapoznanie uczestników z osobą prowadzącą i innymi uczestnikami ze swojej grupy.

Przedstaw się i powiedz, kim jesteś, wskazując zwłaszcza swoje związki ze szkołą/nauczycielami oraz doświadczeniem dotyczącym programowania – zaczniesz w ten sposób budować sobie relację z grupą i wiarygodność jako eksperta/ekspertki.

Zaproś uczestników do krótkiej rundki, w czasie której każdy powie 2-3 zdania na swój temat – jak ma na imię, czego uczy, z jakiej szkoły/miejscowości przyjechał/przyjechała. Celem tej części jest zadbanie o to, żeby każda osoba się wypowiedziała i zrzuciła ciężar zabrania głosu publicznie w obcym środowisku.

2. Zapoznanie uczestników z programem szkolenia. Ustalenie zasad obowiązujących podczas szkolenia, przypomnienie, o czym była część online.

Przedstaw cel szkolenia i przypomnij logistyczne aspekty całego programu (ile szkoleń, jak wygląda cały program szkolenia; że w ramach ścieżki edukacyjnej nauczyciele przygotowują po dwa własne scenariusze zajęć z wykorzystaniem istniejących lub własnych e-materiałów oraz min. 2 lekcje, które przeprowadzą na podstawie tych scenariuszy; że lekcje mają być obserwowane przez innych nauczycieli, w tym członka kadry zarządzającej szkołą itd. Scenariusze lekcji powinny być opracowane samodzielnie i opublikowane na jednej z wolnych licencji; wykorzystywane gotowe e-materiały powinny być przywoływane z zachowaniem autorstwa lub opracowane samodzielnie).

Rozdaj karteczki samoprzylepne i poproś uczestników, żeby zapisali na nich zarówno swoje oczekiwania, jak i obawy związane z tym projektem (ich udziałem w nim); poproś, żeby każde oczekiwanie/obawa zapisana była na oddzielnej karteczce; jak napiszą, niech przykleją na flipczart; jak przykleją, odczytaj wszystkie zapiski i się odnieś, komentując oczekiwania – czy uda się je spełnić, czy nie, oraz obawy – jeśli są to sprawy trudniejsze, możesz zwrócić się do grupy z pytaniem, jak wspólnie możecie zadbać o to, żeby obawy nie znalazły odzwierciedlenia w rzeczywistości.

Umów się z grupą na zasady pracy na szkoleniu (kontrakt) – zapiszcie je na flipczarcie i powieś w widocznym miejscu – to również element budowania zaufania i poczucia bezpieczeństwa dla uczestników; te zasady będą obowiązywać na całym szkoleniu, jeśli potrzeba mogą być wspólnie modyfikowane. Przykładowe zasady znajdziesz w podręczniku trenerskim projektu.

Zawieś na ścianie pusty flipczart i zapowiedz, że jest to miejsce, gdzie będziecie zapisywać nazwy różnych programów, aplikacji, stron www, które pojawią się w czasie szkolenia; będzie to dla uczestników ściągawka-inspiracja na przyszłość, bo

nie wszystko będzie omówione na szkoleniu; umów się z grupą, że wszyscy odpowiadacie za zapełnianie tego flipczartu.
Podziel uczestników na pary i zaproś do dyskusji o tym, co było w części online.
Zadaj pytania pomocnicze: Czego nowego się nauczyli, co było dla nich znane, która część była rozwinięciem wcześniej posiadanych umiejętności? Poproś kilka osób o podzielenie się wnioskami z dyskusji na forum.

Część warsztatowa

3. Stworzenie prostej aplikacji konsolowej umożliwiającej obliczenie statystyk sprawdzianu w środowisku programistycznym Eclipse

Czas trwania: **290 minut**

- Zaproś uczestników do uruchomienia środowiska programistycznego Eclipse oraz utworzenia nowego projektu. Omów poszczególne elementy projektu, utwórz klasę główną projektu - czas trwania: **15 minut**
- Przedstaw koncepcję aplikacji oraz pojęcia i zagadnienia wykorzystywane w czasie jej tworzenia - czas trwania: **15 minut**
- Omów zagadnienie pobierania danych od użytkownika na przykładzie wczytania imienia i nazwiska ucznia oraz jego oceny za sprawdzian - czas trwania: **60 minut**
 - Pobieranie danych od użytkownika z poziomu konsoli (klasa Scanner)
 - Krótkie omówienie mechanizmu obsługi błędów (konstrukcja try - catch) na przykładzie błędnego wczytania oceny ucznia
 - Wyświetlanie danych
- Zainicjuj rozmowę na temat rozszerzenia funkcjonalności aktualnej wersji aplikacji o możliwość wczytania większej ilości uczniów - czas trwania: **45 minut**
 - Wczytanie wielu uczniów - zastosowanie tablic
 - Duplikacja kodu odpowiedzialnego za wczytywanie danych ucznia oraz jego oceny
 - Rozmowa na temat wad duplikacji kodu - braku elastyczności rozwiązania oraz konieczności wielokrotnych zmian/poprawiania błędów
 - Rozwiązanie problemu duplikacji kodu - zastosowanie pętli
 - Przedstawienie różnych typów pętli - for, foreach, while, do while
 - Wczytanie docelowej liczby uczniów
 - Sprawdzenie czy wczytana docelowa liczba uczniów jest wartością poprawną
 - omówienie wyrażeń logicznych oraz instrukcji warunkowych

- Obliczanie średniej ocen - czas trwania: **45 minut**
 - Omów podstawowe operacje arytmetyczne
 - Omów operacje na liczbach całkowitych vs liczbach zmiennoprzecinkowych
- Wyświetlenie ucznia z najlepszą oraz najgorszą oceną - czas trwania: **30 minut**
- Zainicjuj rozmowę na temat przejrzystości i łatwości zrozumienia kodu powstałej aplikacji - czas trwania: **50 minut**
 - Porusz zagadnienie refaktoryzacji
 - Zapytaj jak wykorzystać funkcję do wydzielenia funkcjonalności odpowiedzialnych za: wczytanie liczby uczniów, wczytanie danych uczniów oraz ich ocen, wyświetlenie średniej oraz najlepszego i najłabszego ucznia pod względem oceny ze sprawdzianu
- Podsumowanie procesu projektowania oraz tworzenia aplikacji - czas trwania: **30 minut.**

Zapytaj uczestników jak się czuli podczas wykonywania zadań. Co miało wpływ na ich zaangażowanie, a co im przeszkadzało? Jakie znaczenie w procesie uczenia się odegrały popełniane błędy? Czy były one uczące?

- Dodatkowe zadanie dla chętnych:
 - Rozszerzenie tematu wczytywania danych o wczytywanie danych z pliku
 - Omówienie koncepcji obiektowości
 - Krótkie omówienie tematu systemów kontroli wersji z praktycznym opisem GIT-a.

Zalety korzystania z systemu kontroli wersji w kontekście pracy wielu osób nad jednym projektem - praca na różnych gałęziach, łączenie gałęzi, rozwiązywanie konfliktów.

Możliwość powrotu do wybranej wersji w każdej chwili.

Zabezpieczenie przed utratą kodu źródłowego.

Możliwość śledzenia postępu prac.

Przedstawienie repozytorium kodu GitHub oraz opublikowanie kodu źródłowego stworzonej w trakcie zajęć aplikacji przy użyciu GIT-a z poziomu środowiska Eclipse.

Zaproponowanie wykorzystania systemu kontroli wersji przez uczniów w projektach grupowych.

1.2. Budujemy bazę

4. Stworzenie aplikacji MS Access realizującej wybrany przez uczestników problem biznesowy np. dziennik szkolny itp.

Czas trwania: **320 minut**

- a. Przedstaw uczestnikom koncepcję całego zadania, następnie podziel je na etapy tak, jak poniżej - czas trwania: **60 minut**
 - Wspólny wybór problemu biznesowego, którego dotyczyć będzie stworzona aplikacja
 - Każdy z uczestników tworzy na kartce koncepcyjny model bazy danych
 - Dyskusja nad propozycjami - wyłaniany jest końcowy schemat koncepcyjny bazy danych
- b. Uczestnicy uruchamiają program MS Access oraz tworzą projekt pustej bazy danych - czas trwania: **15 minut**
- c. Omów z uczestnikami **sposoby tworzenia tabel, definiowania typów pól oraz relacji**, a następnie poproś uczestników o odwzorowanie ustalonego wcześniej koncepcyjnego modelu **bazy danych w MS Access** - czas trwania: **60 minut**
- d. Zapoznaj uczestników z tematem formularzy - czas trwania **80 minut**
 - Pokaż uczestnikom jak utworzyć główny ekran aplikacji, z którego będzie możliwa nawigacja do pozostałych części aplikacji
 - Pokaż, jak tworzyć oraz edytować formularze
 - Omów podstawowe właściwości formularzy
 - Omów formularze tworzenia oraz edycji rekordów
 - Zrób wprowadzenie do języka SQL - proste filtrowanie danych
 - Pokaż jak obsługiwać przekierowania użytkownika z ekranu głównego do poszczególnych formularzy - przekazywanie parametrów
 - Poproś uczestników, aby utworzyli wybrane formularze - np. dodawania uczniów, dodawania sprawdzianów, edycji uczniów itd.
- e. Omów temat tworzenia kwerend w kontekście projektowania raportu - czas trwania: **80 minut**
 - Tworzenie projektu raportu
 - Różne metody tworzenia kwerendy - przy użyciu kreatora kwerend, kodu SQL
 - Wybrane zagadnienia związane z językiem SQL - pobieranie danych, filtrowanie danych, grupowanie danych, funkcje agregujące, złączenia.

Wzmianka o operacjach zmieniających stan bazy danych - dodawanie, aktualizowanie, usuwanie danych

- Wykorzystanie kwerendy w raporcie
- Uczestnicy tworzą raport obrazujący statystyki związane z problemem biznesowym, którego dotyczy aplikacja - np. raport uczniów wraz ze średnią ocen oraz najlepszą i najgorszą oceną

f. **Podsumowanie i dyskusja: 30 minut**

Zapytaj uczestników o to, jak się czuli podczas wykonywania zadań. Co miało wpływ na ich zaangażowanie, a co im przeszkadzało? W jaki sposób są ułożone zadania w poszczególnych aplikacjach? jakie znaczenie odgrywa popełnianie błędów w tych aplikacjach?

Jako podsumowanie dyskusji, przypomnij zasady uczenia się dorosłych, zaprezentuj cykl Kolba, podkreśl znaczenie błędów, dzięki którym uczymy się. Zwróć uwagę na udoskonalanie kodu (skręcanie), jako elementu cyklu Kolba.

5. Tworzenie scenariuszy zajęć dla uczniów.

Czas trwania: **30 minut**

Poprowadź rozmowę z uczestnikami, zachęcając ich do utworzenia własnych scenariuszy lekcyjnych na wybrany temat dotyczący omawianych na szkoleniu zagadnień według wzoru z załącznika nr 1. Zwróć uwagę na kwestię praw autorskich i licencji (scenariusze powinny być autorskie i udostępniane na otwartej licencji). Uczestnicy mają czas do końca 2. modułu szkolenia na stworzenie scenariuszy (po jednym na 1 i 2 cel podstawy programowej) oraz uzyskanie ich akceptacji od osoby prowadzącej. Na podstawie tych scenariuszy będą prowadzić 2 lekcje obserwowane w szkole (moduł 3). Zajęcia mogą być przeprowadzone przez nauczyciela/nauczycielkę dopiero po akceptacji scenariusza przez trenera/trenerkę. Ustal z grupą termin oddania scenariuszy.

Podkreśl, że scenariusz powinien zawierać szczegółowy opis przebiegu aktywności zaplanowanych podczas zajęć oraz ramy czasowe poszczególnych fragmentów zajęć (załącznik 1). Scenariusz należy umieścić na platformie projektu. Scenariusze powinny zawierać linki do znalezionych w sieci lub własnych e-materiałów, z których będą korzystać uczniowie. Jeden z tych materiałów musi być przygotowany z wykorzystaniem platformy epodreczniki.

Uczestnicy wypełniają metryczkę scenariusza zajęć i umieszczają go na dysku Google umożliwiając komentowanie innym uczestnikom.

6. Omówienie z uczestnikami funkcjonalności platformy projektu Lekcja:Enter

Czas trwania: **30 minut**

Prowadzący prosi uczestników o przypomnienie struktury, funkcjonalności i zawartości platformy projektu, m.in. gdzie można znaleźć lektury tematyczne, podstawę programową.

Prowadzący prosi uczestników o zalogowanie się na platformie projektu, a następnie omawia wszystkie funkcjonalności dostępne dla uczestników (fora, materiały, umieszczanie własnych scenariuszy zajęć itp.).

Podsumowanie modułu 1: **10 minut**

- Poproś uczestników, żeby przypomnieli, czym się zajmowaliście podczas tego szkolenia.
- Zaproś uczestników do refleksji na koniec - podzielenia się jednym zdaniem, co było dla nich najważniejsze podczas tego pierwszego szkolenia albo, jeśli wyczujesz w grupie gotowość, do czego czują się zainspirowani po pierwszym szkoleniu?
- Rozdaj hand-out „Prawo autorskie a e-zasoby”. Wyjaśnij, że materiał ten będzie przydatny podczas tworzenia scenariuszy i e-zasobów na dalszym etapie szkolenia, a także w codziennej pracy nauczycieli.

(5) **Potrzebne materiały (hand-out, dodatkowe instrukcje, opisy itp.,)**

- Flipchart, kartki A4, flamastry, karteczki samoprzylepne, masa mocująca lub taśma klejąca
- Prawo autorskie
- Załącznik 1 – wzór scenariusza lekcji
- Cykl Kolba, uczenie się dorosłych
- Przykłady relacji w bazach danych
- Rodzaje pętli z przykładami
- Instrukcje warunkowe
- Wyrażenia logiczne
- Instrukcje do zadania 3 i zadania 4

(6) **Priorytety i inne praktyczne wskazówki dla trenerów**

Po tej części szkolenia nauczyciel/nauczycielka zna:

- narzędzia programistyczne - wybrane środowiska informatyczne oraz aplikacje wspomagające proces tworzenia oprogramowania,
- podstawowe zagadnienia związane z programowaniem - instrukcje wejścia/wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne, funkcje,
- problem utrzymania przejrzystości kodu i związane z tym pojęcie refaktoryzacji,
- koncepcję obiektowości w programowaniu (opcjonalne/dla chętnych),
- repozytorium kodu Github (opcjonalne/dla chętnych),
- wybrane narzędzie kontroli wersji (GIT) oraz zalety, a także możliwości wykorzystania w procesie edukacyjnym (opcjonalne/dla chętnych),
- koncepcję relacyjnych baz danych - tabele, pola, relacje, klucz główny, klucz obcy,
- wybrane operacje języka SQL - pobieranie danych, filtrowanie danych, grupowanie danych, funkcje agregujące, złączenia.

Nauczyciel/ nauczycielka potrafi:

- korzystać z wybranych środowisk informatycznych umożliwiających tworzenie różnego rodzaju aplikacji - Eclipse, Netbeans, Android Studio, MS Access,
- tworzyć proste aplikacje konsolowe w języku Java wykorzystujące instrukcje wejścia/wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne oraz funkcje,
- dostrzegać problemy związane z utrzymaniem zduplikowanego kodu i proponować odpowiednie rozwiązania tego rodzaju problemów,
- stworzyć konto w repozytorium kodu Github oraz opublikować na nim kod źródłowy przy użyciu narzędzia kontroli wersji GIT z poziomu środowiska informatycznego Eclipse (opcjonalne/dla chętnych),
- zaprojektować koncepcyjny model bazy danych realizujący wybrany problem biznesowy,
- stworzyć bazę danych wykorzystującą tabele, pola oraz relacje,
- wygenerować oraz edytować formularz działający na bazie danych przy użyciu oprogramowania MS Access,
- stworzyć kwerendę używając kreatora kwerend oraz kodu SQL,
- zaprojektować raport wykorzystujący odpowiednią kwerendę.

(7) Przykłady dobrych praktyk

Koncepcja odwróconej lekcji - <https://paderewski.lublin.pl/edukacja-i-inspiracje/pl/flipped-clasroom-czyli-tajemnice-odwroconej-lekcji>

(8) Literatura – przeczytaj koniecznie!

Samouczek programisty - <http://www.samouczekprogramisty.pl/>

(9) E-zasoby

- Wprowadzenie do programowania w języku Java
Kurs Java - <https://javastart.pl/baza-wiedzy/darmowy-kurs-java>
- Praktyczny tutorial MS Access -
https://www.youtube.com/watch?v=hFn_P40yNRg
- Kurs Programowania Java -
https://play.google.com/store/apps/details?id=com.ionicframework.kursprogramowania938724&hl=en_US

(10) Narzędzia, w tym aplikacje mobilne

Sprzęt:

- Laptopy, komputery stacjonarne
 - System operacyjny - Microsoft Windows 7/8/10 (32-bit lub 64-bit) lub inny kompatybilny
 - Pamięć operacyjna - minimum 2 GB, rekomendowane 8 GB
 - Wolna pamięć dyskowa - minimum 8 GB, rekomendowane 12 GB
 - Rozdzielczość ekranu - minimum 1280 x 800
 - Procesor - minimum 1,5 Ghz, rekomendowane 2,5 Ghz
 - **zainstalowane oprogramowanie - Eclipse, Netbeans, MS Access, Android Studio**
- Projektor multimedialny, rzutnik
- Internet
- Podstawa programowa

MODUŁ 2.

Rozumienie, analizowanie i rozwiązywanie problemów

(1) Informacje organizacyjne dla trenerów

Przed zajęciami sprawdź, czy wszystkie komputery mają dostęp do internetu, czy na wszystkich komputerach jest zainstalowane oprogramowanie Netbeans i Android Studio. Pamiętaj, że poziom umiejętności uczestników w zakresie programowania w Javie może bardzo różnicowany, od zera do zaawansowanego. Przygotuj się do pracy na różnych poziomach. Wykorzystaj doświadczenie uczestników zaawansowanych do dzielenia się doświadczeniem, uczenia innych.

(2) Moduł 2

Moduł drugi jest modułem praktycznym. Uczestnicy będą zdobywali lub rozwijali swoje umiejętności programowania w języku Java, wykorzystując do tego środowisko Netbeans. W ramach ćwiczeń wykonają proste aplikacje przeglądarkowe badające pierwszość liczb, ilość wyrazów ciągu, czy NWW i NWD zadanych liczb. Należy pamiętać, że zarówno umiejętności matematyczne jak i programistyczne nauczycieli na tym poziomie mogą się znacznie różnić. Należy podkreślić, że ćwiczymy umiejętności na poziomie podstawowym. Ważnym elementem tego modułu są e-zasoby, które poleca trener/trenerka, ale też te, którymi wymieniają się użytkownicy. Często programowania można uczyć się samodzielnie, korzystając z tutoriali i samouczków. Jest to atut dla tych, dla których prezentowane w kursie zagadnienia będą materiałem zupełnie nowym.

2. moduł szkolenia składa się z dwóch części:

- 2.1. Algorytmy na Javie – 350 minut (7,8 x 45 minut)
- 2.2. Mobilne szyfrowania – 300 minut (6,7 x 45 minut)

(3) Cele modułu

- Zapoznanie uczestników ze środowiskiem szkolenia: osobą prowadzącą, tematyką, wykorzystywanymi narzędziami
- Podstawy teoretyczne rozwiązywania problemów - myślenie komputacyjne, metoda "dziel i zwyciężaj"
- Zapoznanie uczestników z algorytmami rozwiązującymi wybrane problemy:
 - Wykorzystanie e-zasobów do przedstawienia działania algorytmów

- Różne sposoby implementacji algorytmów - metody iteracyjne oraz rekurencyjne, algorytmy zachłanne
- Wizualizacja działania algorytmów przy użyciu aplikacji okienkowych oraz aplikacji mobilnych
- Weryfikacja poprawności działania algorytmu
- Omówienie zagadnienia określania złożoności obliczeniowej algorytmów - notacja wielkiego O

(4) Przebieg szkolenia

2.1 Algorytmy na Javie

CZĘŚĆ STACJONARNA:

Część wstępna - czas trwania: **20 minut**

1. Rundka powitalna według pomysłu trenera **5 minut**
2. Zapoznanie uczestników z planem szkolenia - czas trwania: **5 minut**
3. Rozmowa wstępna dotycząca algorytmów- czas trwania: **10 minut**
Zainicjuj rozmowę na temat algorytmów.

Przykładowe pytania: Co to jest algorytm? Jaka jest struktura algorytmu? Przykłady algorytmów z życia codziennego. Jak uczyć algorytmiki, gdzie znaleźć ciekawe materiały?

Część warsztatowa - czas trwania: **780 minut**

4. **Wprowadzenie do tworzenia aplikacji okienkowych w środowisku Netbeans** - czas trwania: **70 minut**
 - a. Wspólnie z uczestnikami utwórz nowy projekt aplikacji w języku Java
 - b. Pokaż, jak utworzyć główne okno aplikacji
 - c. Przedstaw możliwości okna projektowania aplikacji okienkowej oraz wykorzystania palety kontrolek
 - d. Omów temat obsługi zdarzeń na przykładzie obsługi kliknięcia na przycisk

Podczas pracy upewnij się co jakiś czas czy uczestnicy nadążają za Tobą, na bieżąco odpowiadaj na powstałe pytania

5. Stworzenie aplikacji okienkowej badającej pierwszośc liczby - czas trwania: **90 minut**

- a. Omów algorytmy testowania pierwszości liczby - metoda naiwna, udoskonalenia metody naiwnej. Poproś uczestników o zapisanie tych algorytmów w postaci pseudo-kodu
- b. Zaproś uczestników do utworzenia projektu aplikacji okienkowej badającej pierwszośc liczby
- c. Implementacja naiwnego algorytmu testowania pierwszości liczby
- d. Zainicjuj dyskusję z uczestnikami na temat złożoności obliczeniowej naiwnego algorytmu testowania pierwszości liczby - wprowadź pojęcie złożoności obliczeniowej algorytmów - notacja wielkiego O. Omów kilka prostych przykładów kodu i oblicz wraz z uczestnikami ich złożoności obliczeniowe.
- e. Implementacja oraz określenie złożoności obliczeniowej wybranego udoskonalenia algorytmu naiwnego testowania pierwszości liczby

6. Stworzenie aplikacji okienkowej obliczającej zdefiniowaną przez użytkownika ilość elementów wybranych ciągów - czas trwania: **60 minut**

- a. Powiedz co to jest ciąg Fibonacciego, przedstaw wzór ciągu Fibonacciego, który zostanie zaimplementowany (wersja rekurencyjna i wersja iteracyjna)
- b. Podziel uczestników na dwie grupy - jedna implementuje wersje rekurencyjną, druga iteracyjną
- c. Zaproś grupy do zaprezentowania swojego rozwiązania na forum.
- d. Zainicjuj rozmowę na temat wad oraz zalet obu podejść

7. Algorytm Euklidesa - znajdowanie NWW i NWD - czas trwania: **60 minut**

- a. Pokaż zapis algorytmu w postaci pseudo-kodu bądź w Scratch-u (np. <https://scratch.mit.edu/projects/53860966/>)
- b. Zapytaj uczestników jaki problem rozwiązuje pokazany algorytm
- c. Poproś uczestników o samodzielne zaprojektowanie aplikacji okienkowej umożliwiającej obliczenie NWW lub NWD
(możesz podzielić uczestników na dwie grupy i jedną poprosić o zaimplementowanie algorytmu w wersji iteracyjnej, a drugą w wersji rekurencyjnej).

8. Stworzenie aplikacji okienkowej wyszukiwanie wzorca w tekście metodą naiwną oraz metodą Knutha-Morrisa-Pratta - czas trwania: **60 minut**

- a. Zainicjuj rozmowę na temat wyszukiwania wzorca w tekście (dojście do metody naiwnej). Pokaż złożoność obliczeniową algorytmu naiwnego

- b. Omówi algorytm Knutha-Morrisa-Pratta
- c. Wraz z uczestnikami zaprojektuj główne okno aplikacji, a następnie zaimplementuj algorytm Knutha-Morrisa-Pratta
- d. Przeprowadź analizę złożoności obliczeniowej algorytmu Knutha-Morrisa-Pratta

9. Sortowanie - stworzenie aplikacji okienkowej implementującej sortowanie metodą bąbelkową oraz metodą przez wstawianie. Praktyczna analiza złożoności obliczeniowej algorytmów - czas trwania: **90 minut**

- a. Omów oba algorytmy sortowania (metoda bąbelkowa, metoda przez wstawianie). Możesz użyć materiałów wideo w celu łatwiejszego zrozumienia zasady działania algorytmów. Spróbuj zachęcić uczestników do zabawy ruchowej obrazującej sortowanie liczb.

Sortowanie bąbelkowe: <https://www.youtube.com/watch?v=lyZQPjUT5B4>

Sortowanie przez wstawianie:

<https://www.youtube.com/watch?v=ROaIU379I3U>

- b. Przeprowadź uczestników przez proces tworzenia aplikacji okienkowej umożliwiającej sortowanie zdefiniowanej przez użytkownika ilości liczb przy użyciu metody bąbelkowej lub metody przez wstawianie
- c. Omów złożoność obliczeniową algorytmów sortowania
- d. Poproś uczestników o dodanie do stworzonej aplikacji opcji wizualizacji czasu wykonania algorytmu w zależności od ilości elementów - empiryczne udowodnienie złożoności obliczeniowej algorytmów (wykorzystując bibliotekę JFreeChart)
- e. Podaj przykład algorytmu o optymalnej (logarytmicznej) złożoności obliczeniowej realizującego koncept "dziel i zwyciężaj" np. sortowanie szybkie. Pokaż krótki materiał wideo obrazujący działanie algorytmu. Poproś uczestników aby spróbowali zaimplementować go w istniejącej aplikacji i sprawdzili różnice w szybkości działania trzech zaimplementowanych algorytmów sortowania.

Przykładowa implementacja oraz zamiana wersji rekurencyjnej na nierekurencyjną: <http://strefakodera.pl/algorytmy/algorytmy-sortowania/przerabiamy-algorytm-quicksort-z-wersji-rekurencyjnej-na-nierekurencyjna>

10. Podsumowanie części 2.1 - czas trwania: 15 minut

Podsumuj pierwszą część modułu drugiego.

Rundka niedokończonych zdań:

- 1. Najłatwiejsze dla mnie było...

2. Najtrudniejsze było...
3. Muszę popracować nad...

2.2 Mobilne szyfrowanie

11. Wprowadzenie w zagadnienie tworzenia aplikacji mobilnych na platformę Android przy użyciu środowiska Android Studio - czas trwania: **60 minut**

- a. Pokaż, jak skonfigurować emulator urządzeń mobilnych na komputerze - dodanie wirtualnego urządzenia mobilnego (w razie potrzeby)
- b. Opisz strukturę projektu, projektora aplikacji oraz palety gotowych kontrolek
- c. Przeprowadź uczestników przez proces tworzenia prostej aplikacji mobilnej składającej się z napisu tekstowego, przycisku, listy rozwijalnej oraz obsługi kliknięcia na przycisk

12. Stworzenie aplikacji mobilnej umożliwiającej zmianę reprezentacji liczb pomiędzy trzema wybranymi systemami liczbowymi (np. binarny, dziesiętny, szesnastkowy) - w formie np. odwzorowania strony <https://www.liczby.pl/kalkulatory/konwersja-miedzy-systemami-liczbowymi> - czas trwania: **90 minut**

- a. Omów algorytmy zmiany reprezentacji liczb pomiędzy wybranymi systemami liczbowymi
- b. Poproś uczestników o utworzenie projektu aplikacji, która będzie zamieniała sposób reprezentacji liczb
- c. Przeprowadź uczestników przez proces implementacji zmiany reprezentacji liczb pomiędzy dwoma wybranymi systemami liczbowymi
- d. Poproś uczestników o samodzielną implementację ostatniego systemu liczbowego

13. Kryptografia - projekt aplikacji mobilnej umożliwiającej szyfrowanie oraz deszyfrowanie tekstu algorytmem Cezara oraz metodą przestawieniową - czas trwania: **60 minut**

- a. Zainicjuj rozmowę o wybranych algorytmach szyfrowania – nawiąż do rysu historycznego, struktury, ograniczeń. W tym zadaniu rozpatrujemy tylko słowa zbudowane z wielkich liter alfabetu angielskiego (o kodach ASCII odpowiednio od 65 do 90), o długościach nie większych niż 30 znaków.
- b. Zaprojektuj wraz z uczestnikami wygląd aplikacji

- c. Podziel uczestników na dwie grupy - jedna implementuje algorytm Cezara, zaś druga metodę przestawieniową.

14. Problem wydawania reszty najmniejszą liczbą nominałów - stworzenie aplikacji okienkowej bądź mobilnej (w zależności od preferencji uczestników) - czas trwania: **90 minut**

- a. Zainicjuj rozmowę na temat sposobu rozwiązania problemu - intuicyjne dojście do algorytmu zachłannego
- b. Wraz z uczestnikami zaimplementuj algorytm zachłanny
- c. Pokaż na przykładach ograniczenia algorytmu
- d. Wyjaśnij i pokaż uczestnikom uniwersalny algorytm wydawania reszty najmniejszą liczbą nominałów wykorzystujący programowanie dynamiczne

15. Podsumowanie części drugiej modułu 2: 15 minut

Trener/trenerka krótko podsumowuje to, co zostało omówione w ramach części 2.2 modułu. Uczestnicy wypowiadają po jednej najważniejszej nowej lub rozwiniętej kompetencji, którą nabyli podczas szkolenia.

(5) Potrzebne materiały

- W tej części modułu potrzebne będą tablety lub smartfony z systemem Android. W przypadku braku urządzeń należy skonfigurować emulator urządzeń mobilnych.
- Ciąg Fibonacciego
- Algorytm Euklidesa
- Metoda naiwna i Knutha - Morisa- Pratta
- Przykład algorytmu „Dziel i zwyciężaj”
- Reprezentacja liczb w systemach binarnym, dziesiętnym i szesnastkowym
- Szyfr Cezara

(6) Priorytety i inne praktyczne wskazówki dla trenerów

Po ukończeniu tego modułu nauczyciel/nauczycielka zna:

- wybrane algorytmy - badania pierwszości liczby, zamiany reprezentacji liczb między pozycyjnymi systemami liczbowymi, NWD i NWW, wyszukiwania wzorca w tekście metodą naiwną oraz metodą Knutha-Morrisa-Pratta, szyfrowania tekstu metodą Cezara i przestawieniową, sortowania przez wstawianie i metodą bąbelkową, wydawania reszty najmniejszą liczbą nominałów, obliczania wartości elementów ciągu metodą iteracyjną i rekurencyjną,

- zagadnienia związane z określaniem złożoności obliczeniowej algorytmów - notacja wielkiego O,
- możliwości środowiska Netbeans w zakresie tworzenia aplikacji okienkowych oraz możliwości środowiska Android Studio w kontekście tworzenia aplikacji mobilnej na platformę Android,
- metody implementacji algorytmów - iteracyjną oraz rekurencyjną, ich wady i zalety,
- przykłady algorytmów realizujących koncept "dziel i zwyciężaj",
- pojęcie oraz charakterystykę algorytmu zachłannego.

Nauczyciel/nauczycielka potrafi:

- stworzyć prostą aplikację okienkową w środowisku Netbeans,
- stworzyć prostą aplikację mobilną na platformę Android w środowisku Android Studio,
- zaimplementować poznane algorytmy oraz zwizualizować ich działanie na przykładzie aplikacji mobilnej bądź okienkowej,
- określić złożoność algorytmu przy użyciu notacji wielkiego O oraz zwizualizować ją w aplikacji.

(7) Przykłady dobrych praktyk

<https://pl.khanacademy.org/>

(8) Literatura- przeczytaj koniecznie!

„Strategia nauczania – uczenia się infotechniki” pod redakcją Stanisława Dylaka i Stanisława Ubermanowicza; Fundacja Wolnego i Otwartego Oprogramowania Poznań 2014.

(9) E-zasoby

- Wstępny kurs algorytmów - <https://pl.khanacademy.org/computing/computer-science/algorithms>
- Podróż w krainę kryptografii - <https://pl.khanacademy.org/computing/computer-science/cryptography>
- Materiały wideo wyjaśniające działanie poszczególnych algorytmów:
 - Sortowanie bąbelkowe - <https://www.youtube.com/watch?v=lyZQPjUT5B4>
 - Sortowanie przez wstawianie - <https://www.youtube.com/watch?v=ROalU379I3U>
 - Sortowanie szybkie - <https://www.youtube.com/watch?v=ywWBy6J5gz8>

- Przykład implementacji algorytmu Euklidesa (NWD) w Scratch - <https://scratch.mit.edu/projects/53860966/>
- Kalkulator konwersji między systemami liczbowymi - <https://www.liczby.pl/kalkulatory/konwersja-miedzy-systemami-liczbowymi>
- Strona o algorytmach i strukturach danych - <http://www.algorytm.org/>
- Forum wymiany wiedzy oraz wzajemnej pomocy w rozwiązywaniu programistycznych problemów - <https://stackoverflow.com/>

(10) Narzędzia, w tym aplikacje mobilne

Aplikacja Algorytmy -

https://play.google.com/store/apps/details?id=com.do_apps.catalog_285

MODUŁ 3.

Zajęcia w szkole z własnymi scenariuszami

Metodyczne aspekty kształcenia z wykorzystaniem technologii.

Zajęcia w szkole prowadzone zgodnie z przygotowanymi scenariuszami, w tym co najmniej dwie godziny zajęć obserwowanych (2x 45 min).

Czas niewliczony w czas szkolenia.

(1) Informacje organizacyjne dla trenerów

Każdy nauczyciel/każda nauczycielka w ramach projektu Lekcja:Enter przeprowadzi średnio 4 lekcje, w tym 2 lekcje obserwowane. Okres, w którym będą przeprowadzane lekcje należy ustalić z grupą na pierwszym spotkaniu (moduł 1.1). Również podczas pierwszego spotkania należy omówić zasady zaliczenia obserwacji i sposób ewaluacji zajęć. Wszystkie zajęcia muszą być przeprowadzone przed ostatnim dniem szkolenia.

(2) Moduł 3- opis

Moduł 3 to praktyczne wykorzystanie umiejętności zdobytych podczas szkolenia w praktyce szkolnej. Każdy z uczestników ma za zadanie przeprowadzić średnio 4 lekcje w swojej placówce, w tym 2 lekcje obserwowane, w oparciu o samodzielnie stworzone scenariusze. Moduł ten służy ewaluacji i doskonaleniu pracy własnej uczestników.

(3) Cele modułu

- Przeprowadzenie zajęć z uczniami, których na co dzień uczy dany nauczyciel/dana nauczycielka, według zaprojektowanych scenariuszy.
- Ewaluacja wybranych zajęć, na którą złożą się: opinie z obserwacji przeprowadzonych przez inne osoby, w tym członka zespołu kierowniczego szkoły, ankiety uczniów wypełnione przez nich pod koniec zajęć, samoocena zajęć przez nauczyciela/nauczycielkę.

Uwaga. Przynajmniej dwa takie zajęcia w trakcie szkolenia powinny być obserwowane przez członka zespołu zarządzającego tej samej szkoły, w której pracuje nauczyciel/nauczycielka i który/która bierze udział w szkoleniu.

(4) Przebieg samodzielnej pracy uczestnika w tym module

1. **Przeprowadzenie lekcji według własnych scenariuszy** nauczyciela/nauczycielki opracowanych po pierwszym i drugim module.

Nauczyciel/nauczycielka przeprowadza śr. 4 godziny zajęć z uczniami w tym co najmniej 2 zajęcia są obserwowane. Obserwacja zajęć może być prowadzona przez dyrektora szkoły a także może przybierać inne formy, np. obserwacji koleżeńskiej lub spaceru obserwacyjnego (więcej na ten temat w „Przewodniku i narzędziowniku dla dyrektorów”). Przeprowadzając lekcję, a wcześniej ją planując, należy sprawdzić, czy w pracowni jest zainstalowane potrzebne oprogramowanie.

Termin przeprowadzenia lekcji jest indywidualnie określany dla każdej z grup.

Nauczyciel/nauczycielka prowadzi 2 godziny lekcyjne zajęć z uczniami. Zajęcia są obserwowane przez przedstawiciela/przedstawicielkę kadry kierowniczej szkoły.

2. Ewaluacja przeprowadzonych lekcji .

- 1) Rozmowy z osobami obserwującymi, czyli z dyrektorem lub innymi osobami, jeśli zastosowano obserwacje koleżeńskie lub spacer obserwacyjny. Rozmowa powinna zostać poprowadzona według ustalonych dyspozycji do obserwacji, zawartych w arkuszu obserwacji.
- 2) Analiza wyników ewaluacji przeprowadzonej wśród uczniów na zakończenie lekcji. Sposób ewaluacji powinien być zaplanowany przez nauczyciela w scenariuszu zajęć. Nauczyciel może przeprowadzić krótką ankietę wykorzystując pytania zaproponowane w scenariuszu, w części **Potrzebne materiały**
- 3) Autoewaluacja. Nauczyciel/nauczycielka dokonuje samooceny swoich działań. Może nagrać krótki filmik, w którym dokonuje refleksji - przykładowe pytania znajdują się poniżej (w potrzebnych materiałach).

3. Przygotowanie raportu

Nauczyciel/nauczycielka wypełnia ankietę dot. ewaluacji swoich zajęć (na platformie lekcjaenter.pl). Podczas zajęć kończących cykl szkoleniowy nauczyciel/nauczycielka prezentuje wnioski z ewaluacji pozostałym uczestnikom.

(5) Potrzebne materiały

Przykładowe pytania do ankiety dla uczniów:

(w pytaniach 1-3 należy zastosować skalę od 1 do 5; pytania 4-5 to pytania otwarte)

1. Oceń swoje zaangażowanie na dzisiejszej lekcji
2. Jak oceniasz sposób przekazywania instrukcji, czy były dla Ciebie zrozumiałe?
3. Oceń w jakim stopniu nauka programowania wpływa na Twoje uczenie się?
4. Co w dzisiejszej lekcji było dla Ciebie trudne?
5. Z czym na dzisiejszej lekcji poradziłeś/poradziłaś sobie bez problemu?

Nauczyciel/nauczycielka dokonując refleksji może odpowiedzieć na pytania:

1. Jakie metody pracy/środki dydaktyczne wykorzystałem na dzisiejszej lekcji?
2. Jak reagowali uczniowie na proponowane aktywności – kiedy wykazywali największe, a kiedy najmniejsze zaangażowanie?
3. Jakie wnioski płyną dla mnie z tego doświadczenia?
4. Na co zwrócę uwagę przy planowaniu kolejnych zajęć?
5. Czy chcę coś zmienić, a jeśli tak, to co?
6. Jaki to będzie miało wpływ na moje dalsze działania?
7. Jakimi doświadczeniami podzielę się z innymi nauczycielami?

(6) Priorytety i inne praktyczne wskazówki dla trenerów

Nauczyciel/nauczycielka zna:

- sposoby ewaluacji zajęć edukacyjnych.

Nauczyciel/nauczycielka potrafi:

- przeprowadzić zajęcia na wybrany temat z wykorzystaniem technologii
- przeprowadzić ewaluację swoich zajęć w oparciu o metody i narzędzia on i off-lineowe
- zmodyfikować przygotowane scenariusze z uwzględnieniem wniosków z przeprowadzonej ewaluacji
- przeprowadzić zajęcia do scenariuszy, w których wykorzystuje własne materiały elektroniczne, angażując do tego również uczniów

(7) Przykłady dobrych praktyk

Scenariusze w programie Koduj z Klasą:

<https://kodujzklasa.ceo.org.pl/materialy-i-scenariusze>

Samouczek programisty

<https://www.samouczekprogramisty.pl/kurs-aplikacji-webowych/>

Materiały w projekcie Szkoła z klasą 2.0

<https://www.szkolazklasa.org.pl/materialy/java/>

Materiały ze strony Uniwersytet dzieci

<https://wklasie.uniwersytetdzieci.pl/scenariusz/jak-zaprogramowac-wlasny-szyfrator>

(8) Literatura- przeczytaj koniecznie!

Technologie informacyjno-komunikacyjne na lekcjach Przykładowe konspekty i polecane praktyki. Małgorzata Ostrowska, Danuta Sterna, CEO,

https://glowna.ceo.org.pl/sites/default/files/tik_na_lekcjach_2015_06_02.pdf

(9) E-zasoby

Jak zrobić ankietę wykorzystując formularze Google:

<https://www.youtube.com/watch?v=Ogw5NF-ufT4>

(10) Narzędzia, w tym aplikacje mobilne

- Google Forms
- Platforma projektu Lekcja:Enter
- Platforma epodreczniki

MODUŁ 4.

Omawiamy przeprowadzone zajęcia

Metodyczne aspekty kształcenia z wykorzystaniem technologii – omówienie obserwowanych zajęć w szkole – zaliczenie szkolenia

Czas trwania: **45 min online oraz 235 minut stacjonarnie**

(1) Informacje organizacyjne dla trenerów

CZĘŚĆ ONLINE, czas trwania: 45 minut

Przed szkoleniem uczestnik opracowuje wnioski z ewaluacji swoich zajęć, które wpisuje w ankietę dot. przeprowadzenia lekcji obserwowanych przed zajęciami stacjonarnymi, kończącymi szkolenie (ankietę wypełnia oddzielnie dla każdego z zajęć). Raport powinien zawierać następujące informacje:

- wnioski z przeprowadzonej ewaluacji (obserwacja kierownictwa placówki, wyniki ankiet uczniowskich, autoewaluacja)
- informacje na temat części lekcji/scenariusza, z których przeprowadzenia uczestnik jest najbardziej zadowolony
- informacje na temat części lekcji/scenariusza, które należy poprawić/zmodyfikować
- ew. nagrania powstałe w ramach modułu 3.

(2) Moduł 4 - opis

Moduł 4 jest ostatnim modułem w cyklu szkoleniowym (i musi być w całości przeprowadzony jednego dnia). Z jednej strony jest to podsumowanie merytoryczne i „konsumpcja” całego projektu, z drugiej – ostatnie spotkanie grupy.

W pierwszej części, najdłuższej, tworzysz przestrzeń do podzielenia się z innymi osobami wrażeniami i doświadczeniami z przeprowadzonych lekcji „projektowych”. Masz na to około 115 minut, więc nie jest to długo, zwłaszcza jeśli masz na sali 12 czy 15 osób. W tej sytuacji musisz grupę rozdzielić i pozwolić jej pracować samodzielnie. Ważne, żeby wszystkim uczestnikom stworzyć podobne warunki czasowe i organizacyjne do podzielenia się swoimi doświadczeniami. Każda osoba ma czas na prezentację, a potem grupa może zadawać pytania, komentować, gratulować. Ta sesja ma być konstruktywna i edukacyjna, ale też wzmacniająca.

W drugiej części szkolenia jest czas na indywidualną refleksję uczestników. Sesja pt. „Mój plan rozwoju” przewidziana jest jako moduł refleksyjny, skierowany na indywidualne działania uczestników.

Kolejna sesja ma charakter planistyczny. Uczestnicy zastanawiają się, w jaki sposób mogą wesprzeć dyrekcje swoich placówek we wdrażaniu strategii TIK. Może to przyjąć różną formę, w zależności od potrzeb placówki.

Ostatnia sesja zamyka cały cykl. To okazja na spojrzenie na przebytą drogę edukacyjną, rozwój.

Pamiętaj też o tym, żeby przypomnieć uczestnikom o formalnych warunkach zakończenia udziału w projekcie i spodziewaj się wielu pytań na ten temat.

(3) Cele modułu

1. Zajęcia końcowe szkolenia „Lekcja:Enter!”. Uczestnik/uczestniczka szkolenia opracowuje wnioski z ewaluacji swoich zajęć, które wpisuje w ankietę dotyczącą ewaluacji lekcji przeprowadzonej w szkole, przed zajęciami stacjonarnymi, kończącymi szkolenie.
2. Każdy nauczyciel/każda nauczycielka prezentuje/omawia przeprowadzone lekcje, dzieli się doświadczeniami, opiniami uczniów i osób uczestniczących w obserwacji.

(4) Przebieg szkolenia

1. **Rozpoczęcie szkolenia** – czas trwania: **15 minut**

- Przypomnij uczestnikom, że jest to ostatni zjazd, ale nie oznacza to, że jest on najmniej ważny – wręcz przeciwnie: on podsumowuje cały proces edukacyjny i jest jego zwieńczeniem. Co więcej, stanowi ostatnią okazję, żeby w życzliwym gronie współuczestników porozmawiać o swoich pomysłach, wątpliwościach, dylematach.
- Zaproś uczestników do rundki (wszyscy powinni zabrać głos) i odpowiedzi na pytanie: „Z czym zaczynasz szkolenie? Z jakimi emocjami, myślami?”
- Przedstaw program szkolenia podkreślając, że jego przebieg w bardzo dużym stopniu zależy od uczestników, ich gotowości do dzielenia się – swoimi doświadczeniami i informacjami zwrotnymi.

2. **Jestem po lekcjach, opowiem jak było** – czas trwania: ok. **120 minut**

(biorąc pod uwagę dostępny czas, to przy 13-15 osobach w grupie jest nierealne, żeby się zmieścić. W związku z tym rozważ, czy grupy nie podzielić na pół. Ty wprowadzasz wszystkich w zadanie, ustalasz zasady itd., ale potem w każdej grupie mianowany jest „koordynator sesji”, który pilnuje czasu i ustalonych zasad. Skróci

to czas sesji (7 osób x ok. 15 minut = ok. 105 minut, czyli do 2 godzin szkoleniowych). To też szansa na zachowanie energii u ludzi do końca szkolenia (wysłuchanie 14, nawet ciekawych opowieści, jest wyczerpujące i traci edukacyjny sens, Pilnuj czasu i elastycznie reaguj na sytuację, np. skracając czas wypowiedzi pozostałych osób – np. pytając tylko 3 osoby, a nie całą grupę/podgrupę).

Każdy uczestnik/uczestniczka ma 15 minut, aby opowiedzieć o swojej lekcji, którą przeprowadził/przeprowadziła i poddał/poddała ewaluacji.

Zapisz na flipczarcie strukturę wypowiedzi:

a. czas – max. 15 minut

b. pytania, na które uczestnik może/powinien odpowiedzieć:

- o fakty: Jak przebiegała lekcja? Jakie narzędzia TIK zostały zastosowane? Co po lekcji powiedzieli uczniowie? A co dyrektor/dyrektorka i ew. inne osoby obserwujące lekcję?
- o emocje: Z czego jestem zadowolona/zadowolony? Co poszło inaczej niż było zaplanowane? Co mnie zainspirowało / zaskoczyło pozytywnie? W jaki sposób wykorzystam zebrane informacje? Na jakie trudności napotykałam/em podczas wykonywania zadania?
- o przyszłość: Czego nauczyło mnie to doświadczenie? Jakie wnioski wyciągam z tego doświadczenia?

Po wystąpieniach poproś grupy o komentarze. Pilnuj, żeby było konstruktywnie i wzmacniająco, a nie krytycznie, dołując. (**max. 15 minut na komentarze**)

3. **Mój plan rozwoju - sesja refleksyjno-rozwojowa – 30 minut**

- Poproś uczestników, żeby sięgnęli do swoich notatek i innych materiałów, jakie posiadają, zastanowili się nad doświadczeniem nauki programowania i pomyśleli o sobie jako „nauczycielu/nauczycielce uczącym/uczącej programowania w sposób angażujący uczniów”. Co widzicie?
- Połącz uczestników w pary lub trójki (zdecyduj patrząc na grupę, jej liczebność i czas, jaki pozostał) i zaproś do podzielenia się wnioskami. Osoby słuchające mogą dopytywać, wskazywać ewentualnie inne mocne strony/potencjały osoby, ale nie dokładać deficytów.
- Zbierz uczestników z powrotem na forum i zapytaj, czy ktoś dowiedział się o sobie czegoś ciekawego, czy jakaś konkluzja własna lub towarzyszy w zadaniu otworzyły mu na coś oczy. A może – pojawił się świetny plan? Zapytaj, czy ktoś jest gotów, żeby podzielić się konkretnymi działaniami, jakie podejmie, żeby podnieść swoje kompetencje w zakresie programowania.

4. Rozwój szkoły w kierunku nauki programowania i TIK – 20 minut

Sesja, której celem jest wsparcie dyrektorów we wdrożeniu nauki programowania i TIK.

W zależności od grupy, wybierz jeden z trzech poniższych wariantów. Możesz tę część zrealizować według własnego pomysłu.

1. Zaproś uczestników do dyskusji grupowej, w jaki sposób nauczyciel/nauczycielka może wesprzeć dyrekcję szkoły we wdrażaniu nauki programowania i planu „Aktywna lekcja z TIK” (plan włączania TIK do działań dydaktycznych w szkole). Zapisuj pomysły na flipczarcie przez 15 minut, a pozostałe 5 minut przeznacz na zapytanie kilkorga uczestników, które z tych pomysłów mogliby zastosować u siebie.
2. Zaproś uczestników do pracy indywidualnej, zadaj im pytanie, w jaki sposób widzą swoją rolę we wsparciu dyrekcji szkoły we wdrażaniu nauki programowania i strategii TIK. Daj im 10 minut na pracę, kolejne 10 minut przeznacz na odsłuchanie 3-4 osób z konkretnymi pomysłami.
3. Zaproś uczestników do pracy w grupach według szkół – każdy pracuje nad swoją szkołą. Wspólnie zapisują pomysły na wsparcie dyrekcji we wdrażaniu nauki programowania i strategii TIK (jeśli osób jest więcej z jednej szkoły, to można je podzielić na kilka grup z jednej szkoły; ważne żeby grupy nie były liczniejsze niż 3-4 osoby). Na koniec poproś 1-2 grupy o prezentację.

5. Zakończenie szkolenia i cyklu szkoleniowego – 30 minut

Przypomnij uczestnikom, jakie są formalne warunki ukończenia/zaliczenia szkolenia:

- Umieścić na platformie projektu wszystkie utworzone przez uczestnika/uczestniczkę materiały tj. autorskie scenariusze wraz z towarzyszącymi im materiałami elektronicznymi.
- Umieszczone scenariusze zostaną skomentowane przez osobę prowadzącą szkolenie. Warunkiem ukończenia szkolenia jest ich akceptacja przez trenera/trenerkę.

Powiedz uczestnikom, że bardzo nas interesuje ich opinia o całym programie edukacyjnym i w związku z tym zapraszamy ich do dwóch form podsumowania/ewaluacji. Najpierw tu wspólnie, w formie rundki końcowej, a za chwilę indywidualnie, przez wypełnienie ankiety online - końcowej ankiety oceniającej szkolenie.

Zapowiedz, że część osób (wybranych losowo) zaprosimy do wypełnienia tzw. Post-testu - ankiety badającej przyrost kompetencji cyfrowych po szkoleniu. Ankieta będzie polegała na samoocenie. Pytania będą podobne jak w preteście, który każdy z uczestników wypełniał przed przyjazdem na szkolenie, na platformie lekcjaenter.pl.

Zaproś osoby do rundki podsumowującej. Będziesz już dość dobrze znał/znała grupę, więc sam/sama zdecyduj, jak zadasz pytanie: jeśli grupa jest zgrana, otwarta i refleksyjna, to zapytaj, z jakimi myślami, z jakimi emocjami kończą swój udział w szkoleniu. Jeśli natomiast masz obawy, czy to pytanie nie pozostanie bez odpowiedzi, to możesz zadać bardziej konkretne, np. które elementy szkolenia były dla ciebie najbardziej użyteczne. Albo – jakie będą teraz twoje pierwsze działania, związane z tematem szkolenia, po powrocie do szkoły.

(5) Potrzebne materiały

Flipchart, flamastry

(6) Priorytety i inne praktyczne wskazówki dla trenerów

Nauczyciel/nauczycielka zna:

- sposoby ewaluacji zajęć edukacyjnych

Nauczyciel/nauczycielka potrafi:

- potrafi zaplanować i przeprowadzić ewaluację zajęć wykorzystujących technologie, uwzględniając ankiety uczniów i ewentualne raporty z hospitacji zajęć przez dyrektora,
- analizuje wyniki hospitacji i uczniowskich ankiet oraz wyciąga z nich wnioski dla swoich dalszych działań,
- zmodyfikować przygotowane scenariusze z uwzględnieniem wniosków z przeprowadzonej.

(7) Przykłady dobrych praktyk

(8) Literatura- przeczytaj koniecznie!

O wspomaganiu szkoły podstawowej w zakresie wykorzystywania technologii informacyjno-komunikacyjnych w pracy z uczniami <https://www.ore.edu.pl/wp-content/plugins/download-attachments/includes/download.php?id=3085>

(9) E-zasoby

Na platformie projektu trener/trenerka ma dostęp do:

Scenariuszy zajęć przygotowanych przez nauczycieli i zaakceptowanych przez trenerów, wraz z e-zasobami i ew. materiałami dodatkowymi

(10) Narzędzia, w tym aplikacje mobilne

Projektor, internet, laptopy/komputery

Załącznik 1

Scenariusz zajęć prowadzonych przez nauczyciela – wzór na platformie projektu

1. Imię i nazwisko autora/autorki

<automatycznie>

2. Etap edukacyjny i klasa

1. <lista rozwijana>

2. <lista rozwijana>

3. <lista rozwijana>

4. <lista rozwijana>

5. <lista rozwijana>

3. Przedmiot

[pole z instrukcją] *Wybierz nazwę przedmiotu, w ramach którego można przeprowadzić zaproponowane przez Ciebie zajęcia.*

1. <lista rozwijalna>

2. <lista rozwijalna>

3. <lista rozwijalna>

4. Temat zajęć

<edytowalne>

5. Czas trwania zajęć (w minutach)

[pole z instrukcją] *Zajęcia powinny trwać min. 45 minut*

<lista rozwijalna>

6. Uzasadnienie wyboru tematu

<edytowalne>

7. Uzasadnienie zastosowania technologii

[pole z instrukcją] *Wpisz korzyści w nabywaniu nowej wiedzy i umiejętności przez uczniów; dlaczego użycie technologii jest w tym miejscu lepsze niż tradycyjne metody?*

<edytowalne>

8. Cel ogólny zajęć

<edytowalne>

[pole z instrukcją] *Cel powinien opisywać wiedzę, umiejętności i postawy, które uczniowie będą poznawać i doskonalić w czasie zajęć.*

9. Cele szczegółowe zajęć

[pole z instrukcją] *Wskaż umiejętności z podstawy programowej danego przedmiotu*

1. <edytowalne>
2. <edytowalne>
3. <edytowalne>

10. Metody i formy pracy

<edytowalne>

11. Środki dydaktyczne

<edytowalne>

12. Wymagania w zakresie technologii

<edytowalne>

[pole z instrukcją] *Określ, jaki sprzęt (komputery, tablety, smartfony, roboty itd.) są niezbędne do przeprowadzenia zajęć. Wymień także urządzenia peryferyjne. Zaplanuj minimalną liczbę sprzętu (czy uczniowie mogą pracować z parach / trójkach przy jednym. Opisz, jak powinna wyglądać sala (np. zajęcia z niektórymi robotami wymagają wolnego miejsca na środku, aby rozłożyć matę, czy niezbędna jest pracownia komputerowa), czy potrzebny jest ekran do wyświetlania prezentacji, rzutnik itp. Zaznacz, czy zajęcia wymagają specjalnych ustawień / przepustowości internetu (np. praca z filmem, streaming). Zaznacz, czy potrzebne jest stałe łącze.*

13. Przebieg zajęć

AKTYWNOŚĆ NR 1

[pole z instrukcją] *Doświadczenie (odwołanie się do własnych przeżyć, eksperyment, dyskusja). Nauczyciel może stworzyć nowe doświadczenie (aplikację, zadanie) bądź odwołać się do tego czego już doświadczyli uczniowie. Nauczyciele w tym miejscu dokonuje krótkiego opisu na doświadczenie stanowiące fazę wprowadzającą lekcji.*

Tytuł:

<edytowalne>

Czas trwania:

<edytowalne>

Opis aktywności:

<edytowalne>

AKTYWNOŚĆ NR 2

[pole z instrukcją] *Refleksja-analiza doświadczenia, które było udziałem uczniów. Nauczyciel opisuje swój pomysł na refleksje po zakończeniu doświadczenia, np: Nauczyciel poprzez*

odpowiednie pytania i prowadzenie dyskusji pozwala uczniom analizować co się stało i dlaczego to nastąpiło. Nauczyciel na koniec dzieli się swoimi obserwacjami. Wyciąganie wniosków na przyszłość.

Tytuł:

<edytowalne>

Czas trwania:

<edytowalne>

Opis aktywności:

<edytowalne>

AKTYWNOŚĆ NR 3

[pole z instrukcją] *Teoria- zebranie faktów/ wniosków w ogólną teorię. Na tym etapie nauczyciel umożliwia uczniom przypomnienie sobie lub poznanie teorii, która pozwala wyjaśnić doświadczony i przeanalizowany zadanie lub wyciągnięcie wniosków, na podstawie których uczniowie mogą uogólnić doświadczenie, tworząc własną "teorię" (regułę, procedurę postępowania).*

Tytuł:

<edytowalne>

Czas trwania:

<edytowalne>

Opis aktywności:

<edytowalne>

AKTYWNOŚĆ NR 4

[pole z instrukcją] *Praktyka-wykorzystanie wiedzy z teorii w działaniu. Nauczyciel/nauczycielka przedstawia pomysł praktycznego zastosowania zdobytej wiedzy. Uczniowie pod okiem nauczyciela/nauczycielki sprawdzają swoje umiejętności - czy potrafią zastosować nową wiedzę - a następnie wprowadzają korekty.*

Tytuł:

<edytowalne>

Czas trwania:

<edytowalne>

Opis aktywności:

<edytowalne>

14. Sposób ewaluacji zajęć

<edytowalne>

[pole z instrukcją] *Zaplanuj ewaluację lekcji - w jaki sposób uczniowie udzielą informacji zwrotnej*

15. Tagi <lista rozwijalna>

16. Licencja <lista rozwijalna>

CC BY-NC-SA 4.0 - Uznanie autorstwa-Użycie niekomercyjne-Na tych samych warunkach 4.0 Międzynarodowe (<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.pl>)

CC BY-NC 4.0 - Uznanie autorstwa-Użycie niekomercyjne 4.0 Międzynarodowe (<https://creativecommons.org/licenses/by-nc/4.0/deed.pl>)

CC0 1.0 Universal - Przekazanie do Domeny Publicznej (<https://creativecommons.org/publicdomain/zero/1.0/deed.pl>)

17. Wskazówki dla innych nauczycieli korzystających z tego scenariusza

<edytowalne>

18. Materiały pomocnicze

<możliwość załączenia pliku - standardowe formaty, też edytowalne; ograniczenie wagi i liczby plików>

[box do odhaczenia] Oświadczam, że jestem wyłącznym autorem/wyłączną autorką scenariusza zajęć, który stanowi utwór w myśl ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych oraz że przysługuje mi pełny zakres majątkowych i osobistych praw autorskich. Posiadane przeze mnie prawa nie są ograniczone jakimikolwiek prawami osób trzecich, jestem uprawniony do rozporządzania nimi, przy czym scenariusz może zawierać elementy takie jak zdjęcia czy cytaty, których wykorzystane zostały w ramach tzw. dozwolonego użytku edukacyjnego.