

SCENARIUSZ ZAAWANSOWANEGO E-MATERIAŁU

1. Metryczka materiału

Tytuł materiału	Programowanie elektroniki jest fajne
Numer materiału	VII.17
Autorzy scenariusza	Marzena Krzysztoń, Monika Skucińska, Michał Szymczak
Weryfikacja WCAG	Zespół ekspertów ds. WCAG (Dominika Gaponiuk, Agnieszka Brodowska, Urszula Grygier, Łukasz Mroziński)
Weryfikacja założeń techniczno-informatycznych	Zespół informatyków ds. integrowania e-materiałów pod względem technologicznym (Paweł, Tomaszek, Katarzyna Gagan, Anna Magdziarz-Tomaszek, Grzegorz Kusztełak)
Weryfikacja językowa	Elżbieta Chrabołowska
Rodzaj multimedium	wirtualne laboratorium
Wykorzystanie AR lub VR AR - rozszerzona rzeczywistość VR - wirtualna rzeczywistość	standardowa 2D lub 3D <input type="checkbox"/> AR <input type="checkbox"/> VR
Etap(y) edukacyjny(e), dla których przeznaczony jest materiał	II etap: SP IV-VIII III etap: Liceum / technikum zakres podstawowy
Przedmiot(y), do nauki których przeznaczony jest materiał	fizyka informatyka technika



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



2. Opis materiału

Skrócony opis materiału (abstrakt)

Materiał edukacyjny jest wstępem do nauki elektroniki i programowania, skierowanym do uczniów szkół podstawowych (klasy 7-8) oraz ponadpodstawowych. Jego celem jest rozwijanie umiejętności projektowania i programowania układów elektronicznych z wykorzystaniem płytki Arduino oraz różnego rodzaju czujników. Treści podzielono na poziomy trudności – od prostych projektów z jedną diodą LED po bardziej zaawansowane układy, takie jak system zliczania samochodów wjeżdżających i wyjeżdżających z parkingu. Materiał zawiera bazę co najmniej 50 projektów, które obejmują różne poziomy zaawansowania, od prostych zadań dla początkujących po średnio zaawansowane projekty wymagające integracji kilku czujników i programowania. Uczniowie będą mieli za zadanie zaprojektować układ oraz go programować w języku C++ lub Python. Dodatkowo, materiał umożliwi zapisanie programu w pamięci Arduino, co pozwala na testowanie działania układu w rzeczywistości. Ta praktyczna nauka wprowadza uczniów w podstawy elektroniki i kodowania, rozwijając ich zdolności analityczne i techniczne.

Cel ogólny materiału

Wyjaśnienie podstawowych definicji elementów elektronicznych wraz z przykładowym zastosowaniem danego komponentu (np. dioda, rezystor, kondensator). Zastosowanie podstaw elektroniki w informatyce, podstawy doboru komponentów (np. rezystorów, kondensatorów). Wyjaśnienie, jak budować i testować układy elektroniczne - np. kontrolować czy są zwarcie itp. Wyjaśnienie i zaprezentowanie na przykładach, w jaki sposób programować układy elektroniczne zbudowane z Arduino.

Sposoby realizacji celów:

- Na podstawie schematu układu elektronicznego uczeń zmontuje wirtualny układ, a następnie go zaprogramuje.
- W aplikacji należy uczniowi dać możliwość zmontowania układu i w razie błędów uczeń otrzymuje odpowiedź/informację zwrotną, jaki błąd popełnił i jakie skutki może to nieść ze sobą.
- Należy w materiale stopniować trudności zadań. Zadania najbardziej złożone mają polegać na samodzielnym zamontowaniu układu na podstawie opisu i projektów wcześniej realizowanych.
- Podczas projektowania zadań należy się posłużyć zestawami, dostępnymi na rynku.

Cele z podstawy programowej kształcenia ogólnego możliwe do realizacji za pomocą materiału

Szkoła podstawowa

Technika

Uczeń:

- dokonuje montażu poszczególnych części układów elektronicznych w całość;
- dokonuje pomiarów za pomocą odpowiedniego sprzętu pomiarowego;
- odczytuje i interpretuje informacje zamieszczone w instrukcjach obsługi urządzeń, na tabliczce znamionowej, opakowaniach żywności, metkach odzieżowych, elementach elektronicznych itp.;
- projektuje i konstruuje modele urządzeń technicznych, w tym elektryczno-elektronicznych.

Szkoła ponadpodstawowa

Informatyka(zakres podstawowy)

Uczeń:



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



- projektuje i programuje rozwiązania problemów z różnych dziedzin, stosuje przy tym: instrukcje wejścia/wyjścia (w tym odczyt danych z czujników, wyświetla wyniki w konsoli lub na wyświetlaczu);
- stosuje w programowaniu wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne, funkcje z parametrami i bez parametrów, testuje poprawność programów;
- posługuje się pojęciem natężenia prądu wraz z jego jednostką; stosuje do obliczeń związek między natężeniem prądu a ładunkiem i czasem jego przepływu przez przekrój poprzeczny przewodnika;

Fizyka(zakres podstawowy)

Uczeń:

- posługuje się pojęciem napięcia elektrycznego jako wielkości określającej ilość energii potrzebnej do przeniesienia jednostkowego ładunku w obwodzie; stosuje jednostkę napięcia;
- posługuje się pojęciem pracy i mocy prądu elektrycznego wraz z ich jednostkami; stosuje do obliczeń związki między tymi wielkościami; przelicza energię elektryczną wyrażoną w kilowatogodzinach na dżule i odwrotnie;
- posługuje się pojęciem oporu elektrycznego jako własnością przewodnika; stosuje do obliczeń związki między napięciem a natężeniem prądu i oporem; posługuje się jednostką oporu;
- rysuje schematy obwodów elektrycznych składających się z jednego źródła energii, jednego odbiornika, mierników i wyłączników; posługuje się symbolami graficznymi tych elementów;
- dokonuje pomiarów za pomocą prądu, oporu i innych parametrów prądu;

3. Charakterystyka materiału

Opis zawartości merytorycznej materiału

Materiał będzie realizowany w postaci prostej wirtualnego laboratorium, a każde zadanie będzie skutkowało przejściem na wyższy poziom.

Rozpoczęcie: Wprowadzenie z podstawowymi wskazówkami, jak rozpocząć pracę. Centralny obszar ekranu z listą projektów (grupowane w poziomy trudności). Każdy poziom odpowiada bardziej złożonym układom i zadaniom.

- **Poziom 1.:** Proste zadania (np. zapalenie diody).
- **Poziom 2.:** Średnio zaawansowane projekty (np. czujniki i interakcje z diodami).
- **Poziom 3.:** Zaawansowane projekty (np. zliczanie samochodów, inteligentny dom).

Na każdym poziomie użytkownik powinien wykonać 2 zadania z wymienionej listy:

- Sygnalizacja świetlna
- Ogród - system podlewania kwiatów
- Stacja pogodowa
- Zabezpieczenie gniazda 230V
- Zegar binarny
- Parking - zliczanie samochodów
- Włącznik światła 230V
- Oświetlenie drogowe - światła z czujnikiem poziomu światła
- Zamek do drzwi z wykorzystaniem np. elektromagnesu
- Inteligentny sterownik klimatyzacji i wentylacji
- Kalkulator z podstawowymi działaniami
- Alarm w mieszkaniu (zabezpieczenie okien, drzwi, czujniki ruchu, klawiatura do uzbrajania i rozbrajania alarmu)



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



- Inne projekty (min. 10).

Uczeń wybiera pierwszy dostępny poziom. Na każdym poziomie trudności uczeń otrzymuje kilka zadań do wyboru z danej puli (3 do 5 zadań). Zadania różnią się tematyką i rodzajem komponentów. Po ukończeniu dwóch zadań z danego poziomu trudności uczeń może przystąpić do zadania będącego kompilacją wcześniejszych projektów. Zadania kompilacyjne są automatycznie przydzielane. Wymagają integracji wiedzy z poprzednich projektów. Kolejne poziomy zostają odblokowane po ukończeniu zadań z niższych poziomów. Po zakończeniu danego poziomu użytkownik ma możliwość w dowolnej chwili powrócić do zadań, których wcześniej nie wykonał.

Kluczowe wymagania merytoryczne i dydaktyczne dla Wykonawcy materiału, które muszą zostać uwzględnione

- Wykonawca powinien posiadać wiedzę z programowania oraz tworzenia projektów układów elektronicznych.
- Materiał musi być dostosowany do grupy docelowej.
- Należy stopniować trudności. W tym celu w materiale należy opracować ścieżki, każda ścieżka to zaawansowany układ, np. stacja pogodowa (układ końcowy), a pośrednie zadania to zaprogramowanie poszczególnych czujników, aż do zadania dla zaawansowanych, czyli budowa gotowego projektu.
- Należy wskazywać praktyczne aspekty teorii związanych z prądem elektrycznym.
- Każdy układ należy bardzo dokładnie opisać, powinno być to zrealizowane np. w postaci krótkich filmów.
- Na zakończenie projektu należy umieścić krótki quiz praktyczny, np. na rysunku przedstawiono podłączenie diody czerwonej, pytanie ma dotyczyć doboru rezystora.
- Każdy program z poziomu materiału powinno dać się wgrać do Arduino.

Opis struktury materiału

Po uruchomieniu multimediu użytkownik ma możliwość zaprogramowania jednego z modułów, np. czujników, sekcji numerycznej albo wszystkiego łącznie, projekt polegać będzie na fizycznym zmontowaniu prototypów urządzenia. Do programowania układu użytkownik ma mieć możliwość wyboru języka programowania. Języki, w których można programować, to: C++, Python lub MicroPython, należy też dopuścić metodę przeciągnij-upuść (dostosowany do multimediu i możliwości technicznych). Moduły są niezależne od siebie.

Programowanie układu:

- Możliwość pisania programu w języku C++, Python lub metoda przeciągnij upuść, który steruje działaniem układu.
- Testowanie kodu - możliwość uruchomienia symulacji, sprawdzenia poprawności działania programu.
- Informacja zwrotna- wskazanie błędów w programie.

Testowanie złożonego układu i programu obejmuje:

- Symulowane wgrywanie programu na płytkę Arduino – po kliknięciu przycisku "Wgraj na Arduino".
- na ekranie komunikat typu "Wgrywanie programu na Arduino..." wraz z paskiem postępu
- po zakończeniu tego procesu pojawia się komunikat - "Program wgrany pomyślnie".
- Możliwość uruchamiania symulacji działania układu i możliwość obserwacji działania układu w rzeczywistości.



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



- Możliwość testowania i diagnozowania działania układu za pomocą wirtualnych narzędzi diagnostycznych (np. wirtualnych oscyloskopów, mierników napięcia).
- Symulacja pozwala na dynamiczną reakcję na zmiany w programie lub połączeniach układu, np. zmieniające się dane z czujników będą wpływać na działanie układu.
- Możliwość wprowadzenia zmian i ponowne wgranie kodu w przypadku, gdy układ nie działa zgodnie z oczekiwaniami.
- Po prawidłowym wykonaniu układu i programu komunikat o poprawnym ukończeniu zadania.
- Quiz sprawdzający wiedzę teoretyczną użytkownika z informacją zwrotną.
- Zakończenie poziomu – wyświetlenie wyników, np. na podstawie czasu wykonania, poprawności układu i programu oraz wyników z quizu.
- Zapisanie postępów użytkownika i odblokowanie kolejnych, bardziej zaawansowanych projektów.
- Po ukończeniu danego poziomu możliwość powrotu do ekranu głównego, gdzie odblokowane zostają kolejne poziomy.

Przykładowe podstawowe projekty do wyboru:

- Sygnalizacja świetlna
- Ogród - system podlewania kwiatów
- Stacja pogodowa
- Zabezpieczenie gniazda 230V
- Zegar binarny
- Parking - zliczanie samochodów
- Włącznik świateł 230V
- Oświetlenie drogowe - światła z czujnikiem poziomu światła
- Zamek do drzwi z wykorzystaniem np. elektromagnesu
- Inteligentny sterownik klimatyzacji i wentylacji
- Kalkulator z podstawowymi działaniami
- Alarm w mieszkaniu (zabezpieczenie okien, drzwi, czujniki ruchu, klawiatura do uzbrajania i rozbrajania alarmu)
- i inne projekty (min 10).

Każdy projekt należy podzielić na podprojekty - od najprostszych układów aż do uzyskania gotowego projektu. Niektóre podprojekty mogą być wspólne dla niektórych projektów.

Na każdym poziomie użytkownik wykonuje 2 zadania z wymienionej listy, a następnie wykonuje zadanie, które będzie kompilacją wykonanych wcześniej zadań (czyli każda z gier będzie się składać z 2 poniższych projektów oraz jednego, który jest ich kompilacją, z możliwością dodania np. przycisku lub innego czujnika).

Mechanika materiału

Mechanika materiału

- Realistyczny wygląd stołu prototypowego z elementami elektronicznymi.
- Po prawej stronie ekranu znajduje się panel z elementami, które można przeciągać i upuszczać na obszar roboczy. Elementy te obejmują czujniki, przewody, rezystory, kondensatory i inne komponenty.
- Po kliknięciu na element wyświetla się tabela/lista, która umożliwia konfigurację parametrów, takich jak rezystancja, napięcie czy czułość. Interakcja z elementami odbywa się za pomocą narzędzia „zaznacz, przeciągnij i upuść”.
- Elementy muszą być prawidłowo podłączone, co weryfikowane jest automatycznie. W przypadku błędnego połączenia wyświetlany jest komunikat diagnostyczny.
- Symulacja działania układu - po zmontowaniu układu i napisaniu programu, użytkownik może uruchomić symulację, aby sprawdzić poprawność działania.
- Symulacja pozwala na dynamiczną reakcję elementów – na przykład zmiany parametrów



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



czujników wpływają na działanie układu.

- W przypadku błędów w kodzie lub połączeniach, użytkownik otrzymuje komunikat z opisem problemu.
- Użytkownik powinien mieć możliwość edycji istniejących projektów, aby mógł modyfikować wcześniej stworzone układy, a nie tylko tworzyć je od podstaw
- W aplikacji powinien znaleźć się tryb eksperymentalny, w którym użytkownik może swobodnie testować działanie różnych komponentów bez narzuconych celów.

Fabula i narracja

- Zadania polegają na rozwiązywaniu problemów związanych z projektowaniem i programowaniem układów elektronicznych. Każde zadanie jest przedstawione jako wyzwanie techniczne, np. „Stwórz system sygnalizacji świetlnej na skrzyżowaniu”.
- W przypadku nieukończenia zadania w określonym czasie następuje reset aktywności, co pozwala użytkownikowi na ponowne podejście do problemu.
- Narracja jest prowadzona w sposób jasny i instruktażowy, z krótkimi komunikatami zachęcającymi ucznia do działania oraz wskazówkami, jak rozwiązać problem.

Grafika i estetyka

- Układ interfejsu zapewnia realistyczny widok stołu z prototypowaniem układu elektronicznego, z dokładnym odwzorowaniem wyglądu elementów elektronicznych.
- Podgląd na biurko robocze z możliwością przybliżenia i oddalenia widoku, aby lepiej zobaczyć szczegóły układu.
- Elementy interaktywne, takie jak piny układów, przewody czy porty Arduino, są wyróżnione podczas przeciągania komponentów, co ułatwia ich poprawne podłączenie.

Dźwięk i muzyka

- W tle odtwarzana jest delikatna muzyka, którą można w razie potrzeby wyłączyć.
- Po najechaniu kursorem na elementy na ekranie odtwarzany jest głosowy opis funkcji danego elementu (np. „Dioda LED – element emitujący światło, wymagający rezystora w obwodzie”).
- Dźwięki sygnalizujące poprawne podłączenie elementów oraz efekty dźwiękowe towarzyszące symulacji działania układu (np. kliknięcie przycisku, brzęczyk emitujący dźwięk).

Grafika

Grafika:

- Widok przedstawia wirtualny stół roboczy, na którym znajdują się elementy elektroniczne (czujniki, rezystory, przewody, diody itp.) w odpowiednio uproszczonych, ale rozpoznawalnych kształtach.
- Elementy zachowują proporcje i wygląd zgodny z rzeczywistością.
- Na stole znajduje się uproszczony model komputera z ekranem, na którym użytkownik może programować układ.

Interakcje:

- Użytkownik może przeciągać i obracać elementy myszką, co pozwala na dokładne ich obejrzenie.
- Połączenia układów są przedstawiane za pomocą dynamicznie generowanych linii, które wizualizują przewody, automatycznie dopasowując się do punktów podłączeń.

Ekran programowania:

- Ekran komputera przedstawia interfejs do pisania kodu w językach wskazanych w opisie aplikacji (C++, Python, metoda przeciągnij-upuść).

Realizm podłączeń i uszkodzeń:

- Realistyczność podłączeń układów jest zachowana poprzez jasne wizualne wskazówki, np. kolorowe podświetlenie pinów do podłączenia lub komunikat w przypadku błędnego połączenia.
- Uszkodzenia elementów są wizualizowane w formie prostych efektów, takich jak zmiana koloru elementu na czerwony (np. przepalona dioda).



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



Kolory i oznaczenia wizualne:

- Przewody, piny, i elementy elektroniczne mają różne kolory lub symbole ułatwiające ich rozpoznawanie i poprawne podłączenie (np. czerwony dla zasilania, czarny dla masy).

Widoczność opcji konfiguracji:

- Po kliknięciu na element użytkownik widzi menu konfiguracji (np. rezystancja, napięcie).

Przykładowe inspiracje**Tinkercad** (<https://www.tinkercad.com/>)

Kategoria: Symulacja obwodów i projektowanie CAD.

Opis: Proste narzędzie do modelowania obwodów elektronicznych i programowania Arduino. Pozwala na symulację działania kodu w wirtualnym środowisku.

Inspiracja: Przyjazny interfejs i możliwość testowania układów przed ich fizycznym montażem.

Fritzing (<https://fritzing.org/>)

Kategoria: Symulacja układów elektronicznych.

Opis: Narzędzie do wizualizacji i prototypowania układów elektronicznych, łączące schematy z rzeczywistymi elementami.

Inspiracja: Interaktywny sposób na projektowanie schematów i analizowanie ich działania.

Arduino Web Editor (<https://www.arduino.cc/education/>)

Kategoria: Edytor kodu online dla Arduino.

Opis: Pozwala na pisanie, testowanie i wgrywanie kodu do płytek Arduino bez potrzeby instalowania dodatkowego oprogramowania.

Inspiracja: Integracja programowania w przeglądarce z możliwością podglądu efektów w czasie rzeczywistym.

4. Wymagania WCAG

Opis dostosowania materiału celem spełnienia standardu WCAG

Zaawansowany e-materiał musi uwzględniać założenia uniwersalnego projektowania w edukacji (UDL) oraz być zgodny ze standardami dostępności cyfrowej WCAG obowiązującymi na dzień ogłoszenia naboru, standardem ATAG 2.0 oraz zapisami ustawy z dnia 19 lipca 2019 r. o zapewnianiu dostępności osobom ze szczególnymi potrzebami (Dz. U. z 2019 r. poz. 1696) i ustawy z dnia 4 kwietnia 2019 r. o dostępności cyfrowej stron internetowych i aplikacji mobilnych podmiotów publicznych (Dz.U. z 2019 r. poz. 848). Powinien też uwzględniać dobre praktyki, stosowane w celu zapewnienia wysokiej jakości dostępnych cyfrowo materiałów edukacyjnych.

Użytkownik ze szczególnymi potrzebami, korzystający z przygotowanego zaawansowanego e-materiału, powinien korzystać z mechaniki materiału (menu nawigacyjnego) w taki sam sposób, jak wszyscy użytkownicy. Należy przygotować menu, w którym wybiera on dostosowania materiału do swoich potrzeb. W ramach wybranych dostosowań zaawansowanego e-materiału użytkownik powinien korzystać ze wszystkich zaprojektowanych funkcjonalności. Zaawansowany e-materiał powinien spełniać kryteria dostępu dla technologii dotykowych (np. ekranów dotykowych), dostępności z poziomu klawiatury czy za pomocą zewnętrznych urządzeń wejściowych (np. mysz powiększona), technologii asystujących (np. czytniki ekranu). Poszczególne ułatwienia dostępu oraz ich konfiguracja powinny być dostępne w menu przed uruchomieniem aplikacji. Powinna istnieć również możliwość zapamiętania wybranych przez użytkownika ustawień, tak aby mogła być stosowana przy kolejnych uruchomieniach aplikacji przez użytkownika.



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



Zaawansowany e-materiał powinien spełniać następujące kryteria:

1. umożliwiać użytkownikowi z różnymi potrzebami korzystać z ułatwień dostępu, na wszystkich poziomach i etapach e-materiału;
2. posiadać instrukcję dla użytkowników z różnymi potrzebami, zawierającą informacje o sposobie korzystania z ułatwień dostępu i mechanizmach poruszania się po menu, przygotowaną za pomocą tzw. prostego języka;
3. posiadać rozwiązania z zakresu dostępności, które pozwalają uniknąć QTE lub działań związanych z łączeniem przycisków (uwzględnia ustawienie pozwalające je uprościć lub pominąć/wyłączyć);
4. umożliwiać korzystanie z wirtualnej klawiatury ekranowej (jeśli materiał tego wymaga), którą można sterować za pomocą myszy lub technologii wspomagających, takich jak wzrok lub przełącznik;
5. umożliwiać skorzystanie z pomocy w sytuacjach potencjalnie trudnych, związanych z poruszaniem się po materiale;
6. użytkownik przed skorzystaniem z zaawansowanego e-materiału powinien mieć możliwość zapoznania się tutorialiem objaśniającym, jak korzystać z ułatwień dostępu;
7. mechanika zaawansowanego e-materiału powinna pozwalać na dostęp do wszystkich obszarów interfejsu użytkownika;
8. zaawansowany e-materiał powinien być dostępny za pomocą technologii asystujących, m.in. czytników ekranu, oprogramowania asystującego w technologiach mobilnych.

Jeżeli w materiale będą występowały treści nieinterpretowalne przez technologie asystujące, wykonawca zobowiązany jest zapewnić alternatywę wchodzącą w e-materiał i stanowiącą integralną całość zaawansowanego e-materiału. Bez konsultacji z ekspertami ORE nie dopuszcza się tworzenia alternatywnego (równoległego rozwiązania) dedykowanego osobom z różnymi potrzebami.

Zaawansowany e-materiał musi uwzględniać między innymi potrzeby osób:

- z ograniczeniami wzroku,
- z ograniczeniami słuchu,
- z ograniczeniami ruchu rąk i mobilności,
- z ograniczeniami możliwości poznawczych (związanymi z np. pamięcią, przetwarzaniem informacji, dysleksją),
- z zaburzeniami neurorozwojowymi i psychicznymi (np. spektrum autyzmu, ADHD, stanami lękowymi, epilepsją),
- z zaburzeniami mowy,
- korzystających z czytników ekranu.

Podczas projektowania e-materiału należy uwzględniać różne potrzeby i możliwości użytkowników ze względu na:

Ograniczenia wzroku:

- stosowanie dobrze kontrastujących kolorów, czytelnych rozmiarów i typów fontów, możliwość zmiany i indywidualnego dopasowania przez użytkownika tych elementów;
- stosowanie zawsze widocznego fokusa (przynajmniej częściowo);
- używanie kombinacji koloru, kształtów i tekstu, niestosowanie znaczenia tylko kolorem;
- umieszczanie przycisków i powiadomień w kontekście;
- stosowanie odpowiedniej wielkości, kolorów i rozmieszczenia elementów interfejsu;
- umożliwienie zmiany kolorów dla osób będących daltonistami;
- umożliwienie zmiany wielkości elementów interfejsu;
- używanie dźwięku przestrzennego i rozróżnialnych dźwięków, różnych w zależności od zdarzeń;
- umożliwienie wyboru wyglądu kursora/celownika, zmiany kształtu, wielkości, koloru, jeśli



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



- projektowana mapa interaktywna zakłada bardzo dużo obiektów;
- wyświetlanie istotnych informacji w centrum, na linii wzroku lub możliwość powiększania całości, poszczególnych elementów mapy interaktywnej;
- nawigacja i sterowanie za pomocą klawiatury;
- stosowanie tekstów alternatywnych lub audiodeskrypcji do grafik;
- elementy materiału powinny być duże i łatwe do odróżnienia oraz oddalone od siebie;
- dodanie opisów alternatywnych do obrazów i innych elementów wizualnych, które opisują treści lub funkcje;
- stosowanie dużego kontrastu między istotnymi elementami w materiale;
- użytkownicy niewidomi powinni móc skorzystać z każdej funkcjonalności materiału z poziomu klawiatury.

Ograniczenia słuchu:

- stosowanie prostego języka, niestosowanie figur stylistycznych i idiomów;
- zapewnienie alternatywy tekstowej każdej kluczowej informacji dźwiękowej;
- dodanie napisów i transkrypcji do treści audio i wideo;
- możliwość modyfikacji napisów, zmiana rozmiaru/koloru oraz ich włączania i wyłączania zanim pojawi się dźwięk;
- stosowanie napisów rozszerzonych informujących o dodatkowych dźwiękach i nastroju oraz postaci mówiących;
- stosowanie prostych logicznych i spójnych układów treści;
- zapewnienie możliwości osobnej regulacji dźwięku dla różnych elementów multimedialnych w mapie interaktywnej;
- zastosowanie przełącznika dźwięku mono/stereo w materiałach filmowych i audio (jeśli takie się pojawiają w zaawansowanym materiale).

Ograniczenia ruchu rąk i mobilności:

- umożliwienie w menu materiału ustawienia dużych obszarów klikalnych;
- projektowanie obsługi za pomocą klawiatury i mowy;
- unikanie tworzenia dynamicznych treści, wymagających dużego ruchu myszy;
- nieograniczanie czasu otwarcia okien, wykonania zadań;
- zapewnienie alternatywy dla akcji, wymagających równoczesnych czynności (np. klik zamiast przeciągnij i upuść);
- zapewnienie sterowania przy użyciu prostych kontrolerów.
- unikanie stosowania bardzo precyzyjnych ruchów.

Ograniczenia poznawcze oraz zaburzenia neurorozwojowe i psychiczne:

- używanie prostych, stonowanych barw;
- używanie prostego języka, bez stosowania figur stylistycznych i idiomów;
- używanie krótkich zdań i punktowania;
- używanie wyjaśnienia skrótów;
- tworzenie opisowych przycisków;
- budowanie prostych i spójnych układów treści;
- wyrównanie tekstów do lewej i zachowanie spójnego układu;
- niestosowanie dużych bloków ciężkiego tekstu;
- niestosowanie podkreślania słów, niepochyłania tekstu i pisania wielkimi literami;
- umożliwienie zmiany kontrastu pomiędzy tłem a tekstem;
- niestosowanie ograniczenia czasowego na wykonanie zadania;
- niestosowanie presji czasowej lub związanej z możliwością wykonania tylko jednej próby wykonania zadania.

Ograniczenia związane z korzystaniem z czytników ekranów:

- opisywanie obrazów, stosownie transkrypcji, audiodeskrypcji;
- nieumieszczanie informacji tylko na obrazie lub wideo;
- nadawanie struktury treści i nieoznaczanie jej tylko rozmiarem i rozmieszczeniem tekstu;



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



- stosowanie liniowego logicznego układu;
- umożliwienie sterowania za pomocą klawiatury;
- tworzenie opisowych łączy.

Powyższe wytyczne są jedynie przykładami potrzeb, jakie powinny zostać spełnione przy projektowaniu zaawansowanego e-materiału. Beneficjent konkursowy powinien zapewnić możliwie największą dostępność dla osób z różnymi potrzebami. Rozwiązania związane z zapewnieniem dostępności osobom z różnymi potrzebami Beneficjent konkursowy powinien konsultować z ekspertami ORE na poszczególnych etapach realizacji projektu konkursowego.

5. Wymagania funkcjonalne i techniczne

Kluczowe warunki funkcjonalne dla Wykonawców

Aplikacja musi spełniać wymagania określone w dokumencie „Ogólne wymagania funkcjonalne i techniczne dla e-materiałów”.

Ekran startowy

- Wprowadzenie zawierające podstawowe wskazówki dotyczące rozpoczęcia pracy w aplikacji.
- W centralnej części ekranu znajduje się lista projektów, uporządkowana według poziomów trudności. Każdy poziom zawiera projekty o rosnącym stopniu złożoności oraz bardziej wymagające zadania.

Realistyczna symulacja procesów

- Realistyczne symulacje procesów elektronicznych umożliwiają przeprowadzanie doświadczeń zbliżonych do rzeczywistych.
- Komponenty elektroniczne odwzorowano w proporcjach zbliżonych do rzeczywistych. Użytkownik może manipulować elementami poprzez podnoszenie, obracanie, przybliżanie i konfigurację parametrów (np. zmiana koloru diody, wartości rezystorów).
- Symulacje wiernie odwzorowują działanie komponentów (np. czujników, serwo mechanizmów i wyświetlaczy) reagujących na wgrany kod.
- Warunki środowiskowe (np. temperatura, wilgotność, natężenie światła) można zmieniać za pomocą panelu symulacji.

Nawigacja po środowisku laboratoryjnym

- Zarządzanie przyrządami i materiałami:
 - Komponenty elektroniczne są dostępne w pasku narzędziowym po prawej stronie ekranu, podzielone na kategorie (czujniki, przyciski, wyświetlacze itp.).
 - Po najechaniu na komponenty wyświetlane są ich opisy i właściwości (np. parametry rezystorów, specyfikacje czujników).
- Budowanie układu elektronicznego:
 - Użytkownik dodaje komponenty na stół roboczy metodą przeciągnij-upuść i może łączyć elementy, tworząc funkcjonalne układy elektroniczne.
 - System automatycznie wskazuje błędy połączeń i podaje sugestie ich poprawy.
- Układ ekranu:
 - Stół roboczy - centralna część ekranu, na której użytkownik buduje i testuje układy elektroniczne.
 - Lista komponentów podzielonych na kategorie po prawej stronie ekranu.
 - Panel programowania w dolnej części ekranu umożliwiający pisanie programów.
 - Panel symulacji po lewej stronie ekranu, umożliwiający zmianę parametrów środowiskowych.



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską



- Wirtualne narzędzia diagnostyczne, dostępne w górnym pasku narzędziowym, obejmują oscyloskopy, multimetry i inne narzędzia diagnostyczne.

Dostosowanie do różnych poziomów trudności

- Trzy poziomy trudności:
 - Poziom 1: Proste zadania, np. zapalenie diody.
 - Poziom 2: Projekty średniozaawansowane, np. wykorzystanie czujników.
 - Poziom 3: Zaawansowane projekty, np. budowa inteligentnego domu.
- Progresja zadań - użytkownik odblokowuje kolejne poziomy po ukończeniu poprzednich.
- Po zakończeniu poziomu użytkownik może powrócić do niezrealizowanych zadań oraz wziąć udział w opcjonalnych quizach sprawdzających wiedzę.
- Podsumowanie wyników zawiera czas wykonania zadań, poprawność układów i wyniki quizów.

Rejestrowanie wyników i analiza danych

- Użytkownicy mogą zapisywać całe projekty (schemat układu + kod) i wracać do nich w dowolnym momencie.
- Dostępna jest opcja eksportu programu oraz generowania raportów z wynikami eksperymentów.

System oceny i feedbacku

- System oceniania powinien dostarczać szczegółową informację zwrotną osobno dla każdego elementu projektu: układ elektryczny, poprawność kodu, optymalizacja kodu.
- System wskazuje błędy w programie, udziela informacji o poprawnym ukończeniu zadania i podsumowuje wyniki quizów.
- Po zakończeniu poziomu użytkownik otrzymuje informację zwrotną na podstawie czasu wykonania, poprawności układów i programu.

Personalizacja przez nauczyciela

- Dostosowanie eksperymentów i zasobów:
 - Nauczyciele mogą wybrać dostępne narzędzia, materiały lub poziomy trudności, aby dostosować laboratorium do określonych celów dydaktycznych.
- Tworzenie własnych scenariuszy:
 - Nauczyciele mają opcję tworzenia lub wybierania scenariuszy, które odpowiadają założeniom lekcji.
 - Nauczyciel powinien mieć możliwość tworzenia własnych układów elektronicznych i udostępniania ich uczniom jako dodatkowe zadania.

Kluczowe warunki techniczne dla Wykonawców

Aplikacja musi spełniać wymagania określone w dokumencie „Ogólne wymagania funkcjonalne i techniczne dla e-materiałów”.

Realizm grafiki i optymalizacja wydajności:

- Wysokiej jakości modele 3D: Modele powinny być realistyczne i zoptymalizowane pod kątem płynności działania na różnych urządzeniach.
- Adaptacyjna jakość renderowania: Możliwość dostosowania jakości grafiki w zależności od urządzenia, co zapewnia płynność działania na starszych urządzeniach.

Bezpieczne przechowywanie i synchronizacja wyników eksperymentów:

- Zapisywanie wyników lokalnie oraz w chmurze: Użytkownicy powinni mieć możliwość zapisywania wyników zarówno lokalnie, jak i w chmurze.
- System odzyskiwania wyników: Funkcja odzyskiwania wyników umożliwiająca użytkownikom powrót do wcześniej przeprowadzonych eksperymentów.



Fundusze Europejskie
dla Rozwoju Społecznego



Rzeczpospolita
Polska

Dofinansowane przez
Unię Europejską

