

# Spectra Engine

An immersive physics simulator with interactive exploration of light and optics.

Silnik fizyki optycznej stworzony na potrzeby konkursu Motorola Cup 2024 przez uczniów z Zespołu Szkół Zawodowych Towarzystwa Salezjańskiego, z pasją i zaangażowaniem w świat fizyki i nowych technologii.

# Spis

1.	Wstęp .....	3
2.	Ważne funkcjonalności .....	3
3.	Obsługa światła .....	4
4.	Odbicia światła .....	4
5.	Soczewki wklęsłe i wypukłe .....	6
6.	Pryzmat .....	7
7.	Bonusowe i kreatywne pomysły .....	8
8.	Informacje końcowe .....	10
9.	Od zespołu .....	11

# 1. Wstęp

Aplikacja, którą zdecydowaliśmy się stworzyć to program służący do symulacji zjawisk optycznych. Do wykonania tego zadania postanowiliśmy wykorzystać możliwości jakie daje silnik **Unity**. Wraz z całym zespołem najpierw pochyliśmy się nad tym jak w teorii działają różne zjawiska optyczne: odbicia, soczewki czy pryzmat. Gdy ustaliliśmy pewne wzory, założenia oraz teorie działania poszczególnych elementów, mogliśmy się zabrać za zamienienie tego czego się dowiedzieliśmy w praktykę i realnie działającą aplikację.

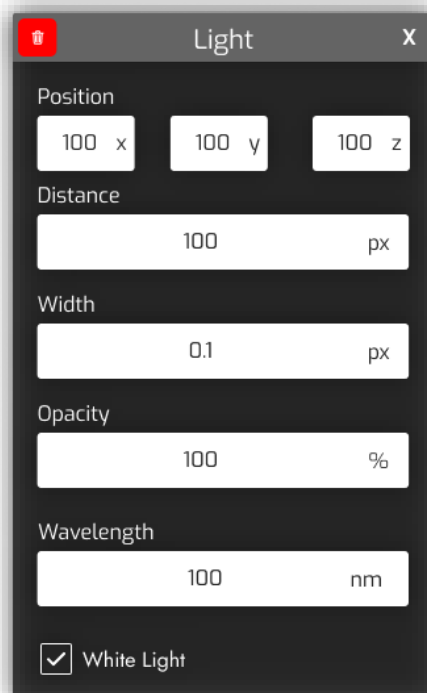
## 2. Ważne funkcjonalności

W tym podpunkcie chcielibyśmy wspomnieć o paru funkcjonalnościach z których możemy korzystać w połączeniu z innymi funkcjami programu. Użytkownik może skorzystać z takich skrótów klawiszowych:

- Gdy przytrzymamy jakikolwiek obiekt będziemy w stanie nim poruszać za pomocą myszki.
- Gdy przytrzymamy obiekt oraz przycisk **R** jednocześnie i zaczniemy ruszać myszką obiekt będzie się obracał w stronę myszki.
- Gdy przytrzymamy obiekt oraz przycisk **S** jednocześnie i zaczniemy oddalać myszkę lub przybliżać, obiekt będzie się skalował w stosunku do odległości myszki od obiektu (z wyjątkiem obiektu **Light**).
- Gdy klikniemy na dany obiekt i naciśniemy przycisk **Delete** znajdujący się na klawiaturze, obiekt zostanie usunięty.
- W trybie 3D myszka jest domyślnie ukryta dlatego na klawiaturze należy nacisnąć przycisk **CTRL** aby się ona pojawiła. Pozwoli nam to na zmianę wartości w panelach do konfiguracji obiektów oraz korzystanie z toolbar'a. Reszta działa identycznie jak w 2D.

### 3. Obsługa światła

Warto wspomnieć o obsłudze samego światła czyli całego „core’a” projektu bo to dzięki wiązce możemy zobaczyć jakiegolwiek efekty działania optyki.



*Światło 1 Parametry wiązki światła*

Możemy zauważyć, że światło składa się z parametrów takich jak:

- Pozycja (X,Y) – możemy manipulować pozycją poprzez wpisanie współrzędnych w celu precyzyjniejszych obliczeń (pozycja jest na bieżąco aktualizowana jeśli jest przenoszona myszką).
- Rotacja (Z) – możemy manipulować obrotem obiektu.
- Szerokość – zmieniamy grubość wiązki ze źródła światła.
- Natężenie – siła z jaką świeci światło.
- Białe światło – gdy włączony światło zamienia się w światło białe, a gdy wyłączony światło świeci kolorem odpowiadającym długości fali.
- Długość fali – wybór długości fali.

### 4. Odbicia światła

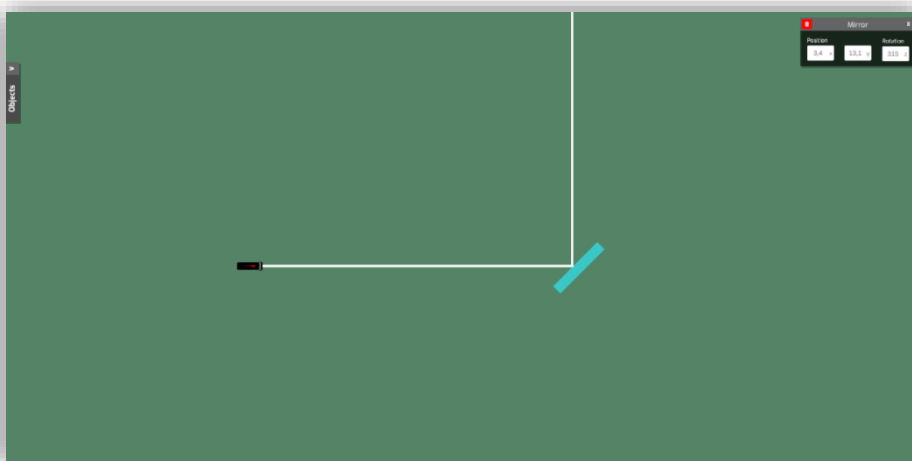
Aby przetestować jak działa zjawisko odbicia światła będziemy potrzebowali lustra oraz źródła światła. Zrobić to możemy poprzez kliknięcie w zakładkę *Objects*. A następnie z naszego panelu wybranie **Mirror** oraz **Light** i późniejsze rozstawienie ich według uznania. Tak jak na obrazkach poniżej. Zwierciadło jest najprostszym obiektem z tego względu możemy manipulować tylko dwoma parametrami czyli jego obrotem oraz pozycją.



*Odbicie 1 Rozwinięcie toolbar*



*Odbicie 2 Wybranie obiektu do stworzenia na planszy*



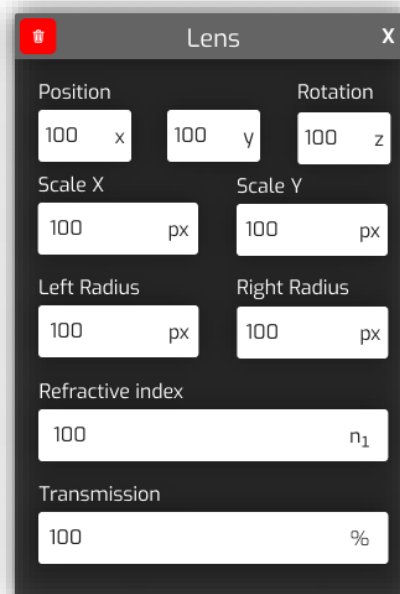
*Odbicie 3 Przykładowe rozmieszczenie światła i zwierciadła*

## 5. Soczewki wklęsłe i wypukłe

Kolejnymi elementami, które możemy umieścić na planszy są soczewki. Do wyboru mamy dwa typy: wklęsła, wypukła. Sposób ich umieszczenia jest identyczny do umieszczenia światła i lustra z punktu *numer 4*. Zauważalna różnica to ta w panelu gdzie możemy zmieniać parametry obiektu.



*Załamanie 1 Załamanie światła dla soczewek*



*Załamanie 2 Parametry jednej z soczewek*

Zauważamy parametry takie jak pozycja i rotacja, działają one identycznie tak jak w ustawieniach lustra czy wiązki światła (są identyczne dla każdego obiektu). Natomiast widzimy też nowe parametry takie jak:

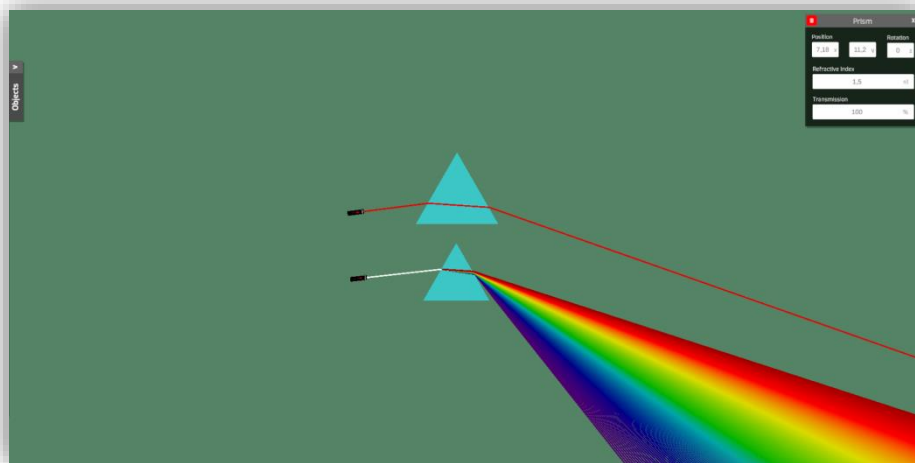
- Skala X, Skala Y – odpowiadają one za szerokość soczewki (X) i wysokość soczewki (Y)
- Lewy i Prawy promień – możemy manipulować promieniem krzywizny soczewki. W soczewce wypukłej oba parametry będą dodatnie natomiast w

wklęsłej będą ujemne. Należy również pamiętać, że dwie krzywe nie mogą być równe 0 jednocześnie.

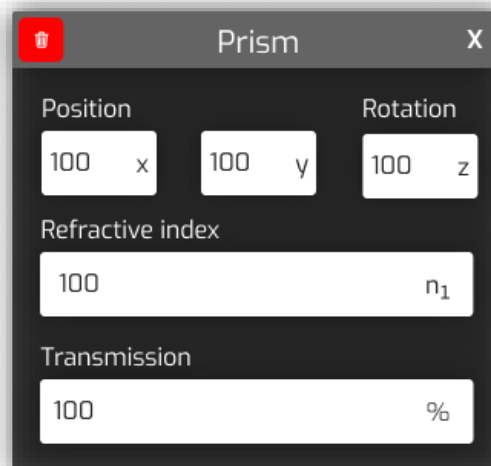
- Współczynnik załamania - jest miarą tego, jak bardzo światło zwalnia i zmienia kierunek, gdy przechodzi przez ośrodek.
- Przepuszczalność – wyrażana w procentach mówi o tym ile światła przedostaje się na zewnątrz a ile zostaje pochłonięte.

## 6. Pryzmat

Ostatnim obiektem jest pryzmat. Poniżej przykładowe rozmieszczenie go na planszy.



*Załączanie 3 Przykładowe rozmieszczenie pryzmatu na planszy*



*Załączanie 4 Parametry pryzmatu*

Analogicznie możemy zmieniać parametry pozycji i rotacji jak w poprzednich przykładach, działają identycznie jak w każdym obiekcie. Współczynnik załamania i przepuszczalność zostały opisane w *punkcie 5*.

## 7. Bonusowe i kreatywne pomysły

Aby aplikacja nie pozostała żmudna, prostolinijna i sztywna, postanowiliśmy wzbogacić ją o ciekawe rozwiązania i dodatki. Między innymi:

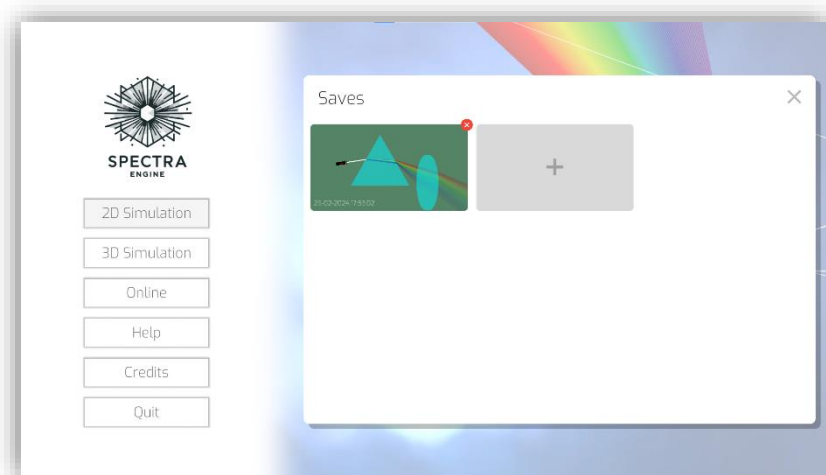
### **Zapis i odczyt sceny**

Użytkownik ma możliwość zapisania aktualnego stanu sceny i zapisania rozłożenia poszczególnych elementów na planszy oraz późniejszego załadowania zapisu. Gdy jesteśmy w wybranej przez nas symulacji pod przyciskiem **ESC** otwiera nam się menu pauzy, gdzie możemy zauważyć przycisk **Save** odpowiedzialny za zapis.



System zapisu 1 Menu pauzy z przyciskiem Save

Jeśli natomiast chcemy wczytać scenę, należy wejść do menu głównego, przyciskiem **Home** lub przy uruchomieniu aplikacji. A następnie wybranie symulacji. Każda symulacja jest zapisywana osobno wraz z datą i uchwyconym zdjęciem stanu sceny pozostawionego przez użytkownika od ostatniego razu. Domyślamy się aby utworzyć nowy zapis należy kliknąć przycisk z plusem, a jeśli chcemy usunąć zapis klikamy w czerwony krzyżyk. (Z informacji technicznych zapisy można zobaczyć w folderze z aplikacją w ścieżce: „Spectra\_Engine\_Data/saves”). Zapis nie uwzględnia custom obiektów wgranych przez użytkownika.

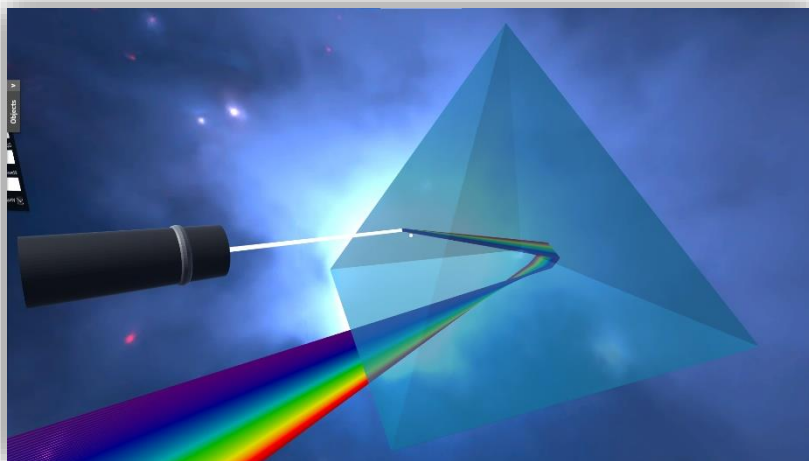


System zapisu 2 Ładowanie zapisanej sceny



## Tryb trójwymiarowy

Kolejną rzeczą jest tryb 3D, użytkownik ma możliwość zobaczenia zjawisk optycznych w świecie trójwymiarowym. Aby nie skłamać jest to feature, który najbardziej nam się podoba z wszystkich rzeczy, które udało nam się do tej pory zaimplementować, swobodne poruszanie się po mapie, z tłem galaktyki i obserwowanie wiązek światła potrafi zrobić wrażenie.

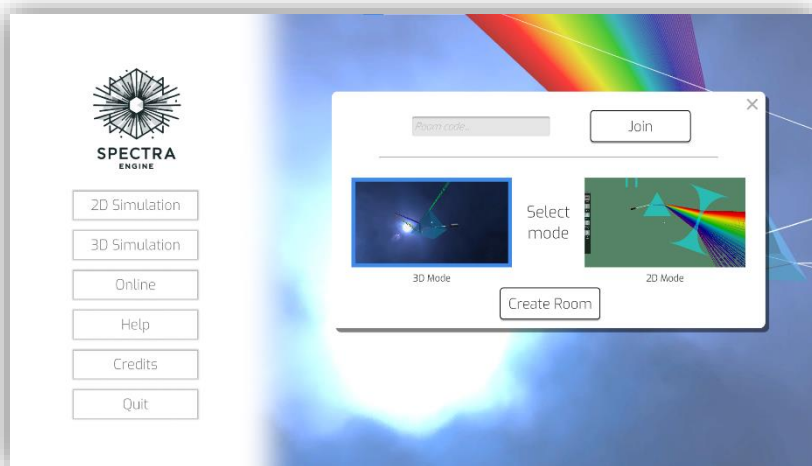


Tryb trójwymiarowy 1 Przykładowa scena 3D

## Tryb online

Ustawiając światła i bawiąc się parametrami obiektów samemu jest po prostu nudno. Dlatego wpadliśmy na pomysł aby wprowadzić tryb współpracy gdzie razem wspólnie ze znajomym czy grupą przyjaciół jesteście w stanie poznawać świat optyki. W menu głównym znajduje się przycisk **Online**. Pojawi się okienko z możliwością wpisania kodu pokoju osoby, która już stworzyła pokój lub możemy taki stworzyć samemu. Do wyboru mamy również dwa tryby **3D** lub **2D**.

Gdy zdecydujesz się na stworzenie pokoju, po wejściu w menu pauzy w prawym górnym rogu będzie znajdować się kod pokoju, który możesz podać innym.



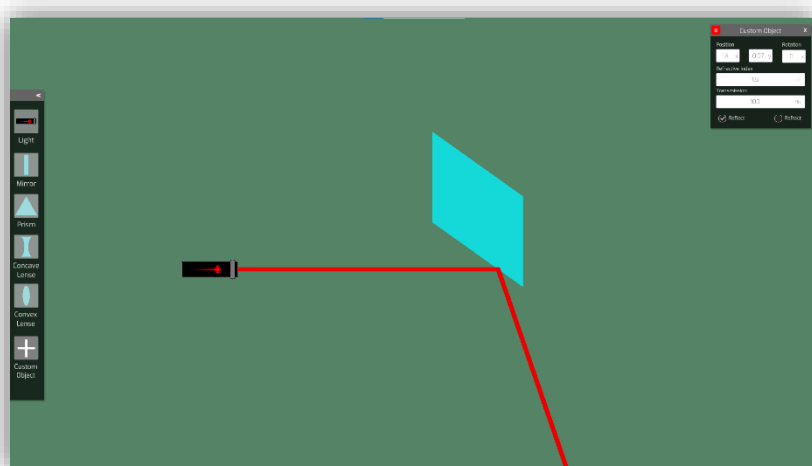
Tryb online 1 Tworzenie i dołączanie do pokoju

## Dowolne kształty

W aplikacji mamy również możliwość importowania dowolnych kształtów zarówno w trybie 3D oraz 2D. Odbywa się to poprzez załadowanie pliku z rozszerzeniem `.obj`.



*Dowolne kształty 1 Dodanie obiektu*



*Dowolne kształty 2 Zmiana ustawień obiektu*

Po zaimportowaniu obiektu jesteśmy w stanie zmieniać mu jego parametry i również przełączać go w tryb **odbicia** lub **załamania**. Czyli nasz obiekt może zachowywać się jak lustro lub soczewka czy pryzmat. Informacje techniczne na temat dodawania własnych obiektów:

- Model nie powinien zawierać świateł oraz kamer
- Wymodelowany obiekt powinien być wycentryowany względem osi X Y Z
- Model powinien składać się z jednego obiektu
- Model musi być w formacie `.obj`

## 8. Informacje końcowe

Dzięki zapoznaniu się użytkownika z tą dokumentacją jest on w stanie wykorzystać pełne możliwości jakie daje aplikacja oraz dowiaduje się do czego służą poszczególne parametry w panelach konfiguracji obiektów.

Program "Spectra Engine" został zrealizowany i opracowany przez zespół uczniów Zespołu Szkół Zawodowych Towarzystwa Salezjańskiego, z pasją i zaangażowaniem w świat fizyki i nowych technologii przy wykorzystaniu takich narzędzi jak silnik Unity.

Projekt powstał w ramach przygotowań do konkursu Motorola Cup 2023, stanowiąc połączenie edukacyjne i technologiczne, mające na celu pogłębienie wiedzy z zakresu optyki, programownia oraz nauk ścisłych.

## 9. Od zespołu

*Cieszymy się, że dostaliśmy możliwość reprezentowania naszej szkoły w konkursie Motorola Science Cup 2023. Jest to dla nas ważne wydarzenie i traktujemy je bardzo poważnie.*

*Projekt, nad którym mieliśmy przyjemność pracować, dał nam nieocenione doświadczenie i pozwolił na praktyczne zastosowanie teorii, której uczyliśmy się na lekcjach. Dzięki niemu mogliśmy przekonać się, jak "suche", podręcznikowe wzory matematyczne i fizyczne ożywają, kiedy zostają wykorzystane w realnych, praktycznych zastosowaniach. To niesamowite doświadczenie zobaczyć, jak nasza praca przekłada się na konkretne rozwiązania i jak teoria łączy się z praktyką w sposób, który wcześniej wydawał się nam tylko abstrakcyjny.*

*Z niecierpliwością czekamy na kolejne wyzwania i projekty, które pozwolą nam dalej rozwijać nasze umiejętności i pasję. Liczymy również, że to nie koniec i zobaczymy się w 3 już ostatnim etapie konkursu.*

*Kacper, lider zespołu.*