|  |
| --- |
| Dead |
| Dead Maze |
|  |
| Dokumentacja |

Autor: Dawid Socha, Robert Pierog

Dead Maze

Spis treści

Spis treści

[Instalacja 2](#_Toc484811247)

[Pobranie niezbędnych plików. 2](#_Toc484811248)

[Uruchomienie gry ( hosting serwera gry ). 2](#_Toc484811249)

[Dołączenie do gry 2](#_Toc484811250)

[Instrukcja gracza 3](#_Toc484811251)

[Poruszanie się 3](#_Toc484811252)

[Zagadki 3](#_Toc484811253)

[Stan gry ( wygrana / przegrana ) 3](#_Toc484811254)

[Wygrana: 3](#_Toc484811255)

[Przegrana: 3](#_Toc484811256)

[Użyte technologie 3](#_Toc484811257)

[Node JS 3](#_Toc484811258)

[Three.js 3](#_Toc484811259)

[MongoDB 3](#_Toc484811260)

# Instalacja

## Pobranie niezbędnych plików.

* Pobierz klienta i serwer bazy danych MongoDB z poniższej strony i zainstaluj.

[***https://www.mongodb.com/***](https://www.mongodb.com/)

* Pobierz serwer NodeJS ze strony poniżej i zainstaluj.

***https://nodejs.org /***

* Pobierz archiwum z grą i wypakuj je w wybrane miejsce na dysku.

## Uruchomienie gry ( hosting serwera gry ).

* Przejdź do folderu gdzie została zainstalowana baza danych MongoDB, trzymając klawisz „shift” naciśnij prawy przycisk myszy i wybierz „Otwórz okno poleceń tutaj”.
* W oknie poleceń wpisz „mongoD” i zatwierdź enter’em.
* Przejdź do folderu gdzie została zainstalowana baza danych MongoDB, trzymając klawisz „shift” naciśnij prawy przycisk myszy i wybierz „Otwórz okno poleceń tutaj”.
* W oknie poleceń wpisz „mongo” i zatwierdź enter’em.
* W oknie poleceń wpisz: use deadmaze i zatwierdź enter’em.
* Przejdź do folderu z grą, trzymając klawisz „shift” naciśnij prawy przycisk myszy i wybierz „Otwórz okno poleceń tutaj”.
* W oknie poleceń wpisz „node server.js” i zatwierdź enter’em.
* Poczekaj aż wyświetli się komunikat, że serwer jest gotowy.
* Otwórz dowolną przeglądarkę internetową i w oknie adresu wpisz: „[TWÓJ\_ADRES\_IP]:3000” i zatwierdź enter’em.
* GRAJ!

## Dołączenie do gry

* Otwórz dowolną przeglądarkę internetową i w oknie adresu wpisz: „[TWÓJ\_ADRES\_IP]:3000” i zatwierdź enter’em.
* GRAJ!

# Instrukcja gracza

## Poruszanie się

W – Ruch do przodu

A – Obrót postaci w lewo

D – Obrót postaci w prawo

## Zagadki

Kliknięcie lewym przyciskiem myszy na drzwi, następnie odpowiedź w okienku i zatwierdzenie odpowiedzi.

## Stan gry ( wygrana / przegrana )

### Wygrana:

Przejście wszystkich poziomów labiryntu szybciej niż drugi gracz.

### Przegrana:

Ukończenie lub nie skończenie labiryntu. Wygrywa drugi gracz.

# Użyte technologie

## Node JS

Jest to środowisko uruchomieniowe zaprojektowane do tworzenia wysoce skalowalnych aplikacji internetowych, szczególnie serwerów www napisanych w języku JavaScript. Umożliwia tworzenie aplikacji sterowanych zdarzeniami wykorzystujących asynchroniczny system wejścia-wyjścia. Jest frameworkiem open source.

Więcej informacji:

<https://pl.wikipedia.org/wiki/Node.js>

## Three.js

Jest to między platformowa biblioteka / API używana do wyświetlania i tworzenia animacji 3D w przeglądarce internetowej. Three.js używa silnika WebGL. Kod źródłowy jest to znalezienia na GitHub’ie (<https://github.com/mrdoob/three.js/> ).

Więcej informacji:

<https://threejs.org/>

## MongoDB

Jest to otwarty, nierelacyjny system zarządzania bazą danych napisany w języku C++. Charakteryzuje się dużą skalowalnością, wydajnością oraz brakiem ściśle zdefiniowanej struktury obsługiwanych baz danych. Zamiast tego dane składowane są jako dokumenty w stylu JSON, co umożliwia aplikacjom bardziej naturalne ich przetwarzanie, przy zachowaniu możliwości tworzenia hierarchii oraz indeksowania.

Więcej informacji:

<https://www.mongodb.com/>