

Zadanie #2 - Wdrażanie mieszkań

Programista w naszej firmie wdraża systemy, buduje aplikacje dla naszych klientów i weryfikuje stan zasobów dostarczanych przez inne zespoły. Dlatego ważne jest czujne oko, aby dostrzec oczywiste błędy. Również bardzo istotna jest wiedza, aby móc sobie z tymi problemami efektywnie radzić. Istotne jest regularne ulepszanie workflow, automatyzacja obecnych systemów oraz rozwijanie aplikacji pod kątem dodatkowych funkcjonalności według potrzeb danego klienta.

Sam proces wdrażania jest w dużej mierze instrukcyjny i częściowo zautomatyzowany, ale błędy ludzkie są tutaj największym ryzykiem. Wymagane jest też pisanie niestandardowych rozwiązań do obecnego core, aby móc zagwarantować jak najlepszy, ale i stabilny produkt.

Twoim zadaniem będzie wdrożenie przykładowych scen mieszkań według przygotowanej instrukcji. Finalnie assety mają trafić do produkcyjnego builda bez żadnych błędów, ale klient jest niecierpliwy i chciałby już zobaczyć postępy w trakcie, kiedy zespół będzie naprawiał zauważone problemy. Dlatego istotne jest również zweryfikowanie wszystkich problemów - na przykład ułomności systemów lub pomyłek członków zespołu.

Nie naprawiaj ręcznie problemów na scenach. Masz tylko zweryfikować te błędy na przeróżne znane Ci sposoby, wypisać je w liście oraz opisać, w jaki sposób je wykryłeś. Masz tutaj pełną dowolność wykorzystania przeróżnych zdolności i technologii.

Podeślij działający projekt, gotowy build oraz plik tekstowy zawierający dokładną listę wszystkich mieszkań wraz ze statusem (błąd/brak błędu) i informacją, co dokładnie na tej scenie jest nie tak. Jeśli jesteś gdzieś w stanie napisać automatyczny fix na to, to jeszcze lepiej (ale poinformuj nas o tym). Powodzenia!

Potencjalne problemy:

- brakujące materiały i meshe
- złe shadery
- minusowa skala obiektów
- nieistniejące referencje do prefabów
- może znajdziesz coś jeszcze?

Forma rozwiązania

Dowolna, liczy się sposób podejścia do problemu oraz kreatywność.

Odpowiedz na pytania

1. Na czym bazowałeś przy szukaniu problemów?
2. Jak można zabezpieczyć się w przyszłości przed błędem ludzkim?
3. Czy użyłeś jakiegoś LLM? Jeśli tak, to gdzie?
 - o w jakim celu?
 - o dlaczego tam?
 - o co poprawiłeś ręcznie?
4. Czego nie zdążyłeś zrobić i co byś jeszcze ulepszył, gdybyś miał więcej czasu?