Figurix

Vytvořte jednoduchou hru, jejímž cílem je v co nejkratším čase najít správnou kombinaci barev a obrázku.

# Pravidla hry

* Každý žeton obsahuje unikátní kombinaci barvy podkladu, orámování a obrázku
* Jsou použity tři barvy a šest obrázků = 54 hracích žetonů
* Na třech hracích kostkách se volí barva podkladu, rámu a konkrétní obrázek
* Hráč musí v co nejkratším čase kliknutím označit žeton se správnou kombinací.

# Návrh herního plánu

Na herní ploše (TokenBoard) jsou „vyskládány“ (náhodně rozmíchané) žetony (Token). Nad nimi je tabulka s herní statistikou (StatusPanel) – čas posledního hledání; počet chybných kliknutí; nejlepší čas hry.

Pod herní plochou jsou „kostky“ (TokenDice) s žetony zobrazujícími (rozfázovaně) aktuálně hledanou kombinaci barev a obrázku.

# Nefunkční požadavky

* Pro řízení logiky hry a získání herních dat je třeba použít třídu figurixGame
* Aplikace bude veškerá data ukládat v Redux store
* Program implementuje návrhový vzor FLUX
* Budou vytvořeny React komponenty na základě schématu v FigurixComponents.pdf
* Komponenty se překreslují jen při změně jejich stavu (např. hození kostkami)

# Postup a další informace

Novou aplikaci lze vytvořit pomocí   
git clone https://github.com/TomasKazda/FigurixKomise.git

Pro realizaci herní logiky použijte třídu figurixGame ze složky /model/

Vhodnou strukturu Redux Store si vytvoříte: const initialState = figurixGame.initGame();

Obnovit stav hry lze např. odesláním celého store do konstruktoru herní třídy:   
new figurixGame(state);

Metodou .setRandomToken() hodíte novou kombinaci na kostkách a zahájíte herní kolo

Metodou .tryTokenSolution(16) realizujete jeden „tip“ – označením žetonu s id = 16

Žeton je vizualizován třídami: <div class="token outcolor2 incolor3 img6">…</div>

Částečný (rozfázovaný) žeton lze zobrazit: <div class="token outcolor incolor2 img">…</div>

Čas je uváděn v milisekundách