

# GeoStar

GeoStar je jednoduchá online webová hra. Hra je postavena na principu boje všichni proti všem. Hráči mají za úkol zasáhnout protivníka a tím získat další bod za zásah.

Hra je přístupná na serveru <http://geostargame.appspot.com> k jejímu načtení je třeba mít Google Account, se kterým se připojíte k serveru. Na uvítací obrazovce hry uvidíte seznam momentálně běžících instancí her. Založení vlastní hry je možné ve spodní části stránky po zadání jména hry. Po zmáčknutí tlačítka "Play!" případně vytvoření nové hry jste přesunut na herní obrazovku. V horní části obrazovky se nachází počítadlo úmrtí a zabití. Na pravé straně se nachází tlačítko pro opuštění hry. Hlavní část stránky zabírá herní plocha. Na herní ploše se pohybují lodičky ve tvaru trojúhelníků. Vaše lodička má barvu bílou a všechny ostatní barvu červenou. Svoji lodičku ovládáte pohybem myši po herní ploše. Lodička vždy sleduje kurzor a přesouvá se na jeho pozici. Pro obranu a útok je lodička vybavena kanónem. Z kanónu se vystřelí stiskem levého tlačítka myši. Kadence kanónu je omezena na 2 výstřely za sekundu. Vaším úkolem je pomocí kanónu sestřelit co nejvíce nepřátel. Po zastřelení nepřítele se vám zvýší počet zásahů na počítadle. Při zásahu cizí lodičkou umíráte a vaše pozice se změní na náhodnou novou.

## Backend:

Serverová část aplikace je naprogramována v jazyce Java a běží na Google App Engine. Program dodržuje rozvržení MVC kde jako model je používána Bean třída, za view slouží soubor JSP a controller obstarává servlet. Aplikace obsahuje dvě view. Hlavní view, které obstarává uvítací obrazovku a herní view, které zobrazuje hru. Komunikace s datovým úložištěm je realizovaná pomocí JPO.

V původní návrhu komunikace hry bylo zamýšleno užití WebSockets. Bohužel GAE tuto technologii nepodporuje. Alternativní a ideálním komunikačním kanálem pro tuto hru by bylo WebRTC, které nepřenáší herní komunikaci přes server, ale pouze mezi klienty (P2P). Tato technologie je, ale stále ve vývoji a proto nedostupná. Při vývoji bylo také otestováno Google Channel API, které ale bohužel využívá služeb AJAX a je značně pomalé. Po těchto zjištěních jsem byl nucen použít pro komunikaci s klienty AJAX.

Všichni klienti posílají na server (/gameserver) každou vteřinu dotaz pomocí AJAX. Dotaz je ve formátu JSON, který obsahuje pozici hráče, vektor pohybu a informaci o nově vypuštěných projektilích. Server podle unikátního klíče nalezne hru, kterou hráč hraje. Aktualizuje data o hráči ve své paměti a hráči zpátky odešle informace o protihráčích a jejich projektilích. Pokud hráč nepodá o sobě informace nejpozději do 3s od poslední zprávy, tak je vyhozen ze hry.

Bohužel real-time hra by potřebovala mnohem větší obnovovací frekvenci zpráv ze serveru a proto dochází k velice častému trhání.

## Frontend:

Uživatelská část aplikace je naprogramována jako klasická webová stránka. Uvítací obrazovka pouze staticky zobrazuje momentálně aktivní hry a formulář pro vytvoření hry nové.

Herní stránka obsahuje canvas do, kterého je vykreslováno pomocí WebGL. Při každé události requestAnimationFrame je nejdříve aktualizován stav hry pomocí delta času. Při aktualizaci dochází k výpočtu nových pozic lodiček a projektilů pomocí jejich směrového vektoru. Tímto způsobem je alespoň částečně kompenzován obrovský lag způsobený spojením pomocí AJAX. Po aktualizaci pozic dojde k vykreslení scény a nakonec pokud uběhla alespoň vteřina od posledního spojení se serverem, tak je odeslána aktualizace pozice a případně i pozice nově vystřelených projektilů. Při aktualizaci také klient ověří jestli se nepovedl zásah cizí lodičky. Ověření zásahu probíhá čistě na straně střílečícího klienta jednoduchou kontrolou jestli se kružnice okolo protivníka protíná s kružnicí okolo vystřeleného projektilu.

Klient reaguje a události myši a podle nich mění směr a cíl lodičky. Při stisknutí myši je vystřelen v 500ms intervalech projektil.

Zprávy ze serveru ve formátu JSON jsou převedeny na objekty pomocí eval. Každá zpráva obsahuje pozice hráčů s jejich unikátními id, servisní data a pozice nově vypálených střel. Servisní data obsahují informaci o skóre hry, timestamp pro ověření posloupnosti zpráv a nakonec informaci jestli byl hráč zabit. Nově vypálené střely se jednoduše přidají do fronty letících střel. Při aktualizaci hráčských dat se kromě přepsání informací také kontroluje jestli není v poli nějaký nový hráč případně nějaký hráč nebyl. Pokud došlo k úmrtí lokálního hráče tak se aktualizují i jeho data podle informací ze serveru. Lokální hráč se normálně neaktualizuje.

Zdrojový kód: <https://github.com/zarubond/GeoStart>

Instance: <http://geostargame.appspot.com/>