

Zadání úlohy

Vytvořte webovou aplikaci, která bude představovat primitivní simulaci aukce. Ve webovém rozhraní bude k dispozici textové pole pro zadání jména uživatele, dále zde bude aktuální cena dražené věci, informace, zda uživatel v aukci vede (*tzn. zda provedl poslední příhoz a pokud by aukce v daném okamžiku skončila, zvítězil by*) a tlačítko pro provedení příhozu.

Uživatel po otevření webu dané aplikace uvidí aktuální cenu dražené věci. Pokud se bude chtít do aukce zapojit, napíše do textového pole své jméno a klikne na tlačítko „přihodit“. V tu chvíli se zvedne aktuální cena o 5 (*aplikace bude umožňovat pouze fixní příhozy, které zvýší cenu o tuto konstantu*) a uživatel se dostane do vedení, tzn. text na řádce „vedu“ se změní na ANO.

Pokud se do aukce zapojí další uživatel a provede příhoz, všem ostatním uživatelům se na webové stránce automaticky zaktualizuje cena a text na řádce „vedu“ se změní na NE (*webová stránka bude muset nějakým způsobem odebírat události ze serveru, aby se uživatel dozvěděl o příhozu jiného uživatele*).

Uživatel může provést další příhoz, i když je aktuálně ve vedení.

Aukce

Uživatel:

Aktuální cena: **100**

Vedu: **NE**

Co (a jak) musíte řešit

- Back-endová část aplikace (*server, který bude zpracovávat požadavky na příhozy a evidovat aktuální cenu a uživatele ve vedení*) bude implementován v Javě pomocí frameworku SpringBoot.
- Front-endová část aplikace může být implementována libovolným způsobem. Webové stránky můžete skládat přímo na serveru (*např. pomocí technologií JSP, Thymeleaf, Freemarker apod.*) a jen lehce si vypomoci Javascriptem pro příjem událostí o příhozech jiných uživatelů. Nebo můžete webové stránky realizovat v nějakém Javascriptovém frameworku typu Angular, React apod. a s back-endovou částí aplikace komunikovat pomocí REST rozhraní.
- Front-end musí reagovat na příhozy jiných uživatelů, tzn. jakmile server zpracuje příhoz uživatele, musí všem ostatním uživatelům rozeslat informaci, na základě které zaktualizují cenu na webové stránce. Toto můžete řešit např. technologiemi WebSockets, Server Side Events apod.
- Aplikace musí umožňovat souběžné používání neomezeným počtem uživatelů, kteří se budou účastnit aukce.

Co nemusíte řešit

- Aplikace bude řešit aukci pouze jedné věci, není třeba řešit dražbu více položek.
- Identita uživatele bude dána jménem zadaným do textového pole, není třeba řešit žádné přihlašování a ověřování uživatelů. Stejně tak není třeba ošetřovat scénáře typu použití stejného jména z více počítačů současně.
- Není třeba řešit zahájení ani ukončení aukce, aplikace by měla umožňovat pouze do nekonečna přihazovat a „předhánět“ se ve vedení v aukci.
- Není třeba řešit perzistenci dat – po vypnutí serveru se všechna data o aukci zahodí.

Příklad chování aplikace

- Otevřeme 2 okna prohlížeče a v obou dvou zobrazíme webovou stránku aplikace.
- V obou oknech bude aktuální cena 100 a ,vedu' s hodnotou NE.
- V okně č. 1 zadáme jméno ,Muf' a klikneme na ,přihodit'.
- Cena v okně č. 1 se zvedne na 105 a ,vedu' se změní na ANO.
- V okně č. 2 se cena také zvedne na 105.
- V okně č. 1 znovu klikneme na ,přihodit'.
- Cena v okně č. 1 se zvedne na 110.
- V okně č. 2 se cena také zvedne na 110.
- V okně č. 2 zadáme jméno ,Puf' a klikneme na ,přihodit'.
- Cena v okně č. 2 se zvedne na 115 a ,vedu' se změní na ANO.
- V okně č. 1 se cena také zvedne na 115 a ,vedu' se změní na NE.