

PŘÍRODOVĚDECKÁ FAKULTA UNIVERZITY PALACKÉHO
KATEDRA INFORMATIKY

BAKALÁŘSKÁ PRÁCE

Online tahová strategie



2010

Filip Dragoun

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval samostatně.

5. ledna 2011

Filip Dragoun

Anotace

Cílem této bakalářské práce bylo vytvořit online tahovou strategii.

Děkuji panu doktoru Večerkovi, za vedení této bakalářské práce, za pomoc při výskytu problémů a za volnost při práci.

Obsah

1. Úvod	1
1.1. Spuštění hry	1
1.1.1. Vytvoření lokální databáze	2
1.2. Popis hry	2
1.3. Objekty ve hře	3
1.3.1. Osada	4
1.3.2. Armáda	5
1.3.3. Špeh	6
2. Popis systému	6
2.1. Use cases diagram	7
2.2. Class diagram	8
2.3. Uživatelské rozhraní	13
2.3.1. Bez vzorové stránky	14
2.3.2. S vzorovou stránkou bez ověření	15
2.3.3. S vzorovou stránkou s ověřením	15
2.4. Sekvenční diagramy	18
3. Databáze	26
3.1. Tabulky	26
3.2. Triggery	29
3.2.1. KontrolaDrevorubec	29
3.2.2. KontrolaDulNaZezezo	29
3.2.3. KontrolaFarma	30
3.2.4. KontrolaKamenolom	30
3.2.5. KontrolaSklad	30
3.2.6. KontrolaSklad	30
3.2.7. VlozeniZpravy	30
3.3. Pohledy	30
3.3.1. ViewArmadaFinance	30
3.3.2. ViewDorucene	31
3.3.3. View_Hrady	31
3.3.4. ViewKasarnaVlastnik	32
3.3.5. ViewOdeslane	32
3.3.6. ViewOsada	33
3.3.7. ViewOsadaFinance	34
3.3.8. ViewSeznamArmad	34
3.3.9. ViewSeznamOsad	35
3.3.10. ViewSeznamSpehu	36
3.3.11. ViewZpravyDorucene	36
3.3.12. ViewZpravyOdeslane	36

3.4.	Procedury	37
3.4.1.	ArmadaNakupAk	37
3.4.2.	ArmadaNakupMoralky	37
3.4.3.	ArmadaNastavAk	37
3.4.4.	ArmadaOsekejSuroviny	38
3.4.5.	ArmadaPresun	38
3.4.6.	ArmadaPresunSurovin	38
3.4.7.	ArmadaPresunVojaku	39
3.4.8.	HracTah	39
3.4.9.	MensiZeDvou	39
3.4.10.	OsadaDrevorubecNastavDelniky	40
3.4.11.	OsadaDrevorubecZmenaUrovne	40
3.4.12.	OsadaDulNaZezezoNastavDelniky	40
3.4.13.	OsadaDulNaZezezoZmenaUrovne	41
3.4.14.	OsadaFarmaNastavDelniky	41
3.4.15.	OsadaFarmaZmenaUrovne	41
3.4.16.	OsadaHradZmenaUrovne	42
3.4.17.	OsadaKamenolomNastavDelniky	42
3.4.18.	OsadaKamenolomZmenaUrovne	42
3.4.19.	OsadaKasarnaVycvicKavalerie	43
3.4.20.	OsadaKasarnaVycvicKopijniky	43
3.4.21.	OsadaKasarnaVycvicLucistniky	44
3.4.22.	OsadaKasarnaVycvicPovozi	44
3.4.23.	OsadaKasarnaVycvicSedlaky	44
3.4.24.	OsadaKasarnaVycvicSermire	45
3.4.25.	OsadaKasarnaZmenaUrovne	45
3.4.26.	OsadaKostelZmenaUrovne	46
3.4.27.	OsadaNastavAk	46
3.4.28.	OsadaOpevneniZmenaUrovne	46
3.4.29.	OsadaPopulace	47
3.4.30.	OsadaPopulaceVolna	47
3.4.31.	OsadaPopulaceVProdukci	47
3.4.32.	OsadaPresunSurovin	47
3.4.33.	OsadaPresunVojaku	48
3.4.34.	OsadaPrirustekSurovin	48
3.4.35.	OsadaPrirustekZlata	48
3.4.36.	OsadaRadniceZmenaUrovne	49
3.4.37.	OsadaSkladZmenaUrovne	49
3.4.38.	OsadaTah	50
3.4.39.	OsadaZvysitPopulace	50
3.4.40.	SpehNakupAk	50
3.4.41.	SpehNakupObratnost	50
3.4.42.	SpehNastavAk	51

3.4.43. SpehPresun	51
3.4.44. SvetClickNaTahZbyva	51
3.4.45. SvetPocetTahu	51
3.4.46. SvetTah	52
3.4.47. VytvorKraje	52
3.4.48. ZpravaPosli	52
3.4.49. ZpravaPosliId	52
4. Seznam tříd	53
4.1. Armada	53
4.2. ArmadaInfo	55
4.3. Armady	56
4.4. Budova	63
4.5. databáze	63
4.6. Dorucenka	66
4.7. Drevorubec	67
4.8. DulNaZelezo	67
4.9. Farma	68
4.10. Finance	68
4.11. Hrad	69
4.12. Jednotka	69
4.13. Kamenolom	71
4.14. Kasarna	72
4.15. Kavalerie	73
4.16. Kopijnik	74
4.17. Kostel	74
4.18. Kraj	75
4.19. Kraje	76
4.20. Lucistnik	78
4.21. NastaveniTahu	78
4.22. Obrance	79
4.23. Opevneni	80
4.24. Osada	81
4.25. OsadaInfo	85
4.26. Osady	86
4.27. Povoz	95
4.28. Produkce	96
4.29. Radnice	96
4.30. Rozloha	98
4.31. Sedlak	98
4.32. Sermir	99
4.33. Sklad	99
4.34. Speh	100

4.35. SpehInfo	101
4.36. Spehove	102
4.37. Svet	105
4.38. Vojak	105
4.39. Zprava	107
4.40. Zpravy	108
5. Administrace	110
5.1. Přímá editace databáze	111
5.2. Změna vlastností objektů	113
6. Uživatelská část	114
6.1. Finance	115
6.2. Osada	116
6.3. Armada	117
6.4. Špeh	119
6.5. Mapa	119
6.6. Zprávy	120
6.7. Průběh bitvy	120
7. Závěr	121

Seznam obrázků

1. Úvod

Vytvoření webové tahové strategické hry, která bude pro tisíce hráčů, hráči mohou do hry vstoupit kdykoli v průběhu hry a předpokládá se, že hra poběží delší dobu (měsíce až roky). Součástí hry bude i program na generování herního světa, kde se dá nastavit jeho velikost, maximální počet hráčů, čas ukončení hry. Hráč se nejprve zaregistruje a pak se bude do hry přihlašovat. Po registraci hráč obdrží jednu osadu na náhodně vygenerovaném místě. Po přihlášení je možno spravovat své osady, vojenské jednotky a pohybovat se po mapě. Hráči si mezi sebou mohou posílat zprávy a suroviny. Aplikace bude vytvořena ASP.NET 2.0 pomocí programu Web Developer 2008 a C#, databáze MSSQL Server 2005.

Důvod vzniku této hry je především v tom, že na internetu neexistuje žádná podobná hra, která by hráčům umožnila volný pohyb po herním světě. Mám tím na mysli že v konkurenčních systémech je možno posílat armádu z jedné osady na druhou, pohyb na mapě není nikterak znázorněn, jen je v nějakém přehledu znázorněn čas, kdy armáda k cíli dorazí. Hráč nemá možnost armádu nijak ovládat, maximálně může armádu z útoku stáhnout zpět.

Tato hra jako jediná nabízí přesun armád kamkoliv na mapě, kde je terén tráva. Pokud se tak rozhodneme zaútočit na cizí osadu, začneme naši armádu přesouvat k cíli. Ale než se k cíli dostaneme, může se náš záměr změnit a armádu pošleme jinam. Pokud bychom tuto situaci chtěli provést v konkurenčních systémech, museli bychom nejprve vrátit armádu zpět, čekat až se vrátí a pak ji poslat na nový cíl.

Hra je optimalizovaná pro webový prohlížeč „Internet Explorer“. U jiných prohlížečů není zaručeno správné zobrazení všech grafických efektů (např. zelené osvětlení vlasních objektů se na mapě v prohlížeči „Mozilla“ neobjeví).

1.1. Spuštění hry

Aby celý herní systém mohl fungovat, je potřeba nejprve zadat spojení k nějaké existující MSSQL databázi (databáze může být úplně prázdná). Toto spojení se zadává v souboru „Bakalarka/App_Code/Databaze.cs“ ve veřejné proměnné „conn“. Poté přepokopírujeme obsah adresáře „Bakalarka“ na webový server s technologií ASP.NET. Na úvod je potřeba vytvořit v databázi všechny potřebné tabulky, trigger, pohledy a uložené procedury. To uděláme následovně.

Na adrese domény načteme stránku „AdminLog.aspx“, objeví se stránka pro přihlášení administrátora. Defaultní přihlašovací údaje jsou, jméno admin, heslo admin. Pokud v databázi neexistuje tabulka „Admin“ přihlášení se nezdaří, ale zároveň se vytvoří nová tabulka „Admin“ s jedním řádkem odpovídajícím defaultním přihlašovacím údajům. Provedeme tedy opětovné přihlášení a dostaneme se na stránku, kde můžeme vytvořit, nebo opravit celý databázový systém a založit novou hru.

Hru lze spustit i ve „Visual Studiu 2008 - Web Developer 2008“ a vyšším. Zvolíme „Open Web Site“ a vybereme adresář „Bakalarka“. V souboru „App_Code/Database.cs“ ve veřejné proměnné „conn“ nastavíme spojení na existující MSSQL databázi (databáze může být úplně prázdná), nebo si vytvoříme lokální databázi s názvem „TestBakalarka“ (spojení na lokální databázi s tímto názvem je v proměnné „conn“ nastaven).

1.1.1. Vytvoření lokální databáze

K vytvoření lokální databáze musíme mít nainstalovaný „Microsoft SQL Server 2005“ a vyšší. Poté spustíme příkazový řádek a napíšeme následující posloupnost příkazů.

1. sqlcmd
2. CREATE DATABASE [TestBakalarka]
3. go
4. exit

Tímto úspěšně vytvoříme prázdnou lokální databázi s názvem „TestBakalarka“.

1.2. Popis hry

Hráč se nejprve zaregistruje, tím se mu vytvoří na serveru hráčský účet. Na začátku dostane každý hráč jednu osadu umístěnou na náhodně vygenerovaném místě. Všechny tyto nové osady budou v základním stavu, to jest bude postaveno jen pár nejdůležitějších budov s vylepšením na nízké úrovni, ale vše dostatečné pro vývoj a rozšíření osady. Kromě řízení vývoje této počáteční osady, bude moct hráč ovládat i jiné objekty, ty si ale musí teprve vytvořit.

Mezi další objekty k ovládání budou patřit armády, špehové a v neposlední řadě i další osady, které může získat obsazením nepřátelské osady.

Takže po prvotní registraci se uživatelé mohou ke svému hernímu účtu přihlašovat. Po přihlášení se první objeví stránka se seznamem všech osad patřících přihlášenému hráči. Každý řádek seznamu bude reprezentovat jednu osadu. Řádek se bude skládat z názvu osady (libovolný název, který si hráč volí sám bez ohledu na duplicitu těchto názvů) a dalšími atributy dané osady.

Kliknutím na řádek reprezentující osadu se dostaneme do přehledového menu vybrané osady. V tomto menu se po řádcích zobrazí jednotlivé budovy osady. Kliknutím na název budovy se dostaneme na stránku, pomocí níž můžeme budovu vylepšovat a pokud to budova dovoluje, (je-li k tomu určena) můžeme stavět i jiné jednotky.

Kromě možnosti zobrazení seznamu všech osad má hráč k dispozici seznam všech armád, špehů přehled financí a dokonce i mapu, na které budou vyznačeny všechny objekty hráče (osady, armády a špehové) žlutou barvou a všechny ostatní (nepřátelské) osady a armády červenou barvou, (nepřátelští špehové zobrazení nebudou).

Seznam všech armád se zobrazí v řádcích, každý řádek reprezentuje jednu armádu. Na řádku je zobrazen název armády (libovolný název, který si hráč volí sám bez ohledu na duplicitu těchto názvů), aktuální poloha, morálka, zkušenosti a počet akčních bodů.

Seznam všech špehů se zobrazí v řádcích, každý řádek reprezentuje jednoho špeha. Na řádku je zobrazen název špeha (libovolný název, který si hráč volí sám bez ohledu na duplicitu těchto názvů), aktuální poloha, obratnost, zkušenost a počet akčních bodů.

K provádění úpravy jakéhokoliv objektu (osady, armády a špeha), jsou potřeba akční body, které má každý objekt své a tyto akční body nemohou objekty mezi sebou sdílet. Každá úprava nějakého objektu, tak kromě potřeby mít na ní potřebné množství surovin, potřebuje také potřebné množství akčních bodů, např. špeh může popojít na mapě jen o počet políček odpovídající vzorci počet akčních bodů děleno počet akčních bodů potřebných k přesunu o jedno políčko nebo stavba budovy, na kterou máme všechny potřebné suroviny, se nemůže provést, pokud nemáme i dostatečný počet akčních bodů, potřebných k postavení dané budovy. Akční body se obnovují vždy po uplynutí jednoho kola. Nevyužité akční body propadají a do dalšího kola se nepřipočítávají.

Svět je to šachovnice skládající se ze čtyř druhů políček (tráva, lesy, voda, hory). Na této šachovnici se nachází jednotlivé objekty hráčů jako jsou osady, armády, špehové. Na jednom políčku se může nacházet více objektů najednou, kromě osady, ta může být na políčku jen jedna, ale mohou tak s ní být objekty typu armáda a špeh.

1.3. Objekty ve hře

Objekty použité v této hře se dají rozdělit do dvou skupin, podle toho zda mohou svou polohu měnit či ne. Do skupiny objektů s nemožností změny polohy patří jediný objekt a tím je osada. Do skupiny objektů s možností pohybu patří objekty armáda a špeh.

Při změně polohy se mohou pohybovat po políčkách typu tráva. Pohyb po políčkách stojí akční body, cena se stanovuje podle nejpomalejší jednotky, kterou objekt sebou nese, tzn. armáda je složená z různých jednotek a cena akčních bodů za pohyb armády po políčku je dána tou nejpomalejší jednotkou v armádě, pohyb špeha je konstantní.

1.3.1. Osada

Tento objekt disponuje těmito vlastnostmi, název, pozice, kapacita občanů, počet občanů, růst populace, zisk osady, počet akčních bodů. Tyto vlastnosti se dají v osadě měnit pomocí výstavby a zlepšování jednotlivých budov. Každá budova dělá něco jiného.

Zisk celé osady se počítá podle nevyužívaných občanů (občané, kteří nic netěží) osady, tak zvanou volnou populaci osady. Tento zisk se přičítá k pokladnici, která je pro všechny osady jednoho hráče společná, je vázaná na hráče, nikoliv na osadu.

Budovy a jejich účel

- **Radnice:** Se zvyšujícím stupněm této budovy, roste maximální ubytovací kapacita osady, zvyšuje se maximální počet akčních bodů osady, zároveň nabízí přehled o stavu populace, spokojenosti v osadě a od toho odvozený i populační růst, dále přehled o těžbě surovin, a aktuální počet akčních bodů tj. kolik bodů ještě zbývá k rozdělení.
- **Kostel:** Zvyšuje spokojenost v osadě, čím větší stupeň kostela, tím více se zvýší spokojenost v osadě a tím větší růst populace.
- **Skladiště:** Každý stupeň této budovy stanovuje horní hranici počtu surovin, které může mít osada k dispozici, neboli, kolik se do skladiště vejde surovin. Čím vyšší stupeň skladiště, tím více surovin pojme.
- **Farma:** Zde se vytváří jídlo. Množství vytvořeného jídla je závislé na počtu dělníků zde hospodařících. Stupeň této budovy určuje maximální počet dělníků zde pracujících, čím vyšší, tím více dělníků může těžit.
- **Dřevorubec:** Zde se těží dřevo. Množství natěženého dřeva je závislé na počtu dělníků zde těžících. Stupeň této budovy určuje maximální počet dělníků zde pracujících, čím vyšší, tím více dělníků může těžit.
- **Kamenolom:** Zde se těží kámen. Množství natěženého kamene je závislé na počtu dělníků zde těžících. Stupeň této budovy určuje maximální počet dělníků zde pracujících, čím vyšší, tím více dělníků může těžit.
- **Důl na železo:** Zde se těží železo. Množství natěženého železa je závislé na počtu dělníků zde těžících. Stupeň této budovy určuje maximální počet dělníků zde pracujících, čím vyšší, tím více dělníků může těžit.
- **Kasárna:** Slouží k výcviku sedláků, kopijníků, šermířů, lučištníků, kavalérie a povozů. Čím vyšší stupeň budovy, tím více se do kasáren vejde vojáků. K výcviku jednotky bude potřeba (kromě surovin a akčních bodů) mít volnou populaci v osadě, z které se rekrutí berou. Takto vycvičené jednotky se stávají součástí domobrany osady a zůstávají

umístěny v kasárnách. Tito vojáci se již nepočítají k populaci osady a k jejich údržbě jsou potřeba peníze, které se odečítají ze společné pokladnice daného hráče. Udržovací poplatek se počítá jako součet udržovacích poplatků všech jednotek.

- Opevnění: Zvyšuje obranu jednotek bránících osadu, čím větší stupeň opevnění tím vyšší zvýšení obrany obránců.
- Hrad: Slouží ke vytváření generálů a špehů. Se zvyšujícím stupněm je možno vytvářet více generálů a špehů.

1.3.2. Armáda

Slouží jednak k obraně a útoku a dále k přesunu surovin mezi osadami. Armáda se skládá z jednoho generála (podle názvu generála se pak jmenuje i celá armáda) a libovolného počtu sedláků, kopijníků, šermířů, lučištníků, kavalérie a povozů. Útočná (obranná) síla celé armády se počítá jako součet útočné (obranné) síly všech jednotek. Provoz armády stojí udržovací poplatek, který se za pochodu (pokud je armáda mimo osadu) dvojnásobný. Udržovací poplatek se počítá jako součet udržovacích poplatků všech jednotek. Tento udržovací poplatek se pak každé kolo odečítá z pokladnice daného hráče.

Armáda se vytváří v hradu, kde se vytvoří generál, dále musí nabrat jednotky a to tak, že generála přesuneme na políčko osady, z které chceme jednotky nabrat a dáme přeskupit armádu. Tím se dostaneme do menu, kde můžeme přesouvat jednotky mezi objekty armáda a osada (respektive kasarna).

Každá jednotka má tyto vlastnosti: Útočná a obranná síla, zátěž (kolik jednotka unese surovin), rychlost (kolik akčních bodů potřebuje jednotka k přesunu o jedno políčko) a udržovací poplatek (žold).

Jednotlivé druhy jednotek

- Generál: Stanovuje morálku celé armády, pokud morálka klesne na nulu a méně, celá armáda se nenávratně rozprchne. Po vytvoření má každý generál morálku 100, morálka se dá zvýšit nákupem za zkušenostní body. Zkušenostní body se získají jen po vyhrané bitvě a počítají se ze zabitých nepřátelských jednotek. Pokud při bitvě budou zabiti civilisti, tak se útočníkovi o tento počet mrtvých civilistů sníží morálka.
- Sedlák: Je to nejslabší útočná jednotka, její výhodou je schopnost unést větší zátěž.
- Kopijník: Je to levná jednotka, která neoplývá velkou silou, má však bonus obranné síly, pokud je při bitvě na straně obránce.
- Šermíř: Levná jednotka, která má bonus útočné síly, pokud je při bitvě na straně útočící armády.

- **Lučičtník:** Tato jednotka se používá jen při první vlně útoku, kdy v řadách nepřítele způsobuje velké ztráty.
- **Kavalérie:** Vyniká svou velkou útočnou silou za každé situace, první vlny se ale neúčastní, do bojů zasahuje až v druhé vlně.
- **Povoz:** Tato jednotka má zanedbatelnou útočnou sílu, slouží pouze k přesunu surovin mezi osadami.

Jeli armáda bez vojáků, je možné armádu skrýt do některé z vlastních osad. Takto skrytá armáda, se pak na mapě nezobrazuje a nedá se na ni útočit. Pokud však bude osada dobytá a obsazena cizím hráčem, všechny armády ukryté v této osadě budou nenávratně ztraceny.

1.3.3. Špeh

Tato jednotka je schopna zjišťovat informace o osadě či armádě, která se nachází na stejném políčku jako špeh. S každou úspěšně provedenou misí rostou zkušenosti špeha, čím větší má zkušenost, tím větší je pravděpodobnost úspěchu mise. Neúspěšná mise může skončit dvěma způsoby. Buď je špeh chycen a zabit, čímž o špeha přijdete, nebo se špehovi podaří utéct, i když misi nesplnil.

2. Popis systému

Systém online tahové strategie je rozdělen do tří částí, které jsou spolu vzájemně provázány.

- **Databázový systém:** Kromě uložení dat provádí i jednodušší funkce, které by měly patřit spíše do aplikační logiky. Důvod vytvoření těchto funkcí je úspora systémového zatížení webového serveru a zrychlení celého systému. Návrh databázových tabulek obsahuje na první pohled zbytečné funkční závislosti jednoho či skupiny atributů, na jednom atributu v téže tabulce. Tato redundance dat je ovšem nezbytná, pro provádění výše zmíněných funkcí. Popis celého databázového systému se nachází v sekci 3.
- **Aplikační logika:** Hlavní část aplikační logiky se nachází v třídách (potřebných k vytvoření objektů ve hře) a statických třídách zajišťujících převážnou většinu funkcionalit celého systému. Seznam všech tříd s popisem všech jejich atributů, vlastností a metod najdete v sekci 4.
- **Uživatelské rozhraní:** Využívá připravených funkcí v třídách (uložených na webovém serveru) a uložených procedurách (uložených v databázovém systému), ke komunikaci s uživatelem. Zobrazováním a skrýváním těchto funkcionalit dopolňuje uživatelské rozhraní částečně i aplikační logiku celého systému. Správný chod celého systému je tak na uživatelském rozhraní závislý. Bližší informace v sekci 2.3..

Základní popis chování celého systému si popíšeme na use case diagramu. Na diagramu tříd si popíšeme hlavně dědičnost a agregaci objektů. Poté si popíšeme procházení webových stránek a nakonec si na sekvenčních diagramech popíšeme podrobněji posloupnosti provádění metod u vybraných funkcionalit systému.

2.1. Use cases diagram

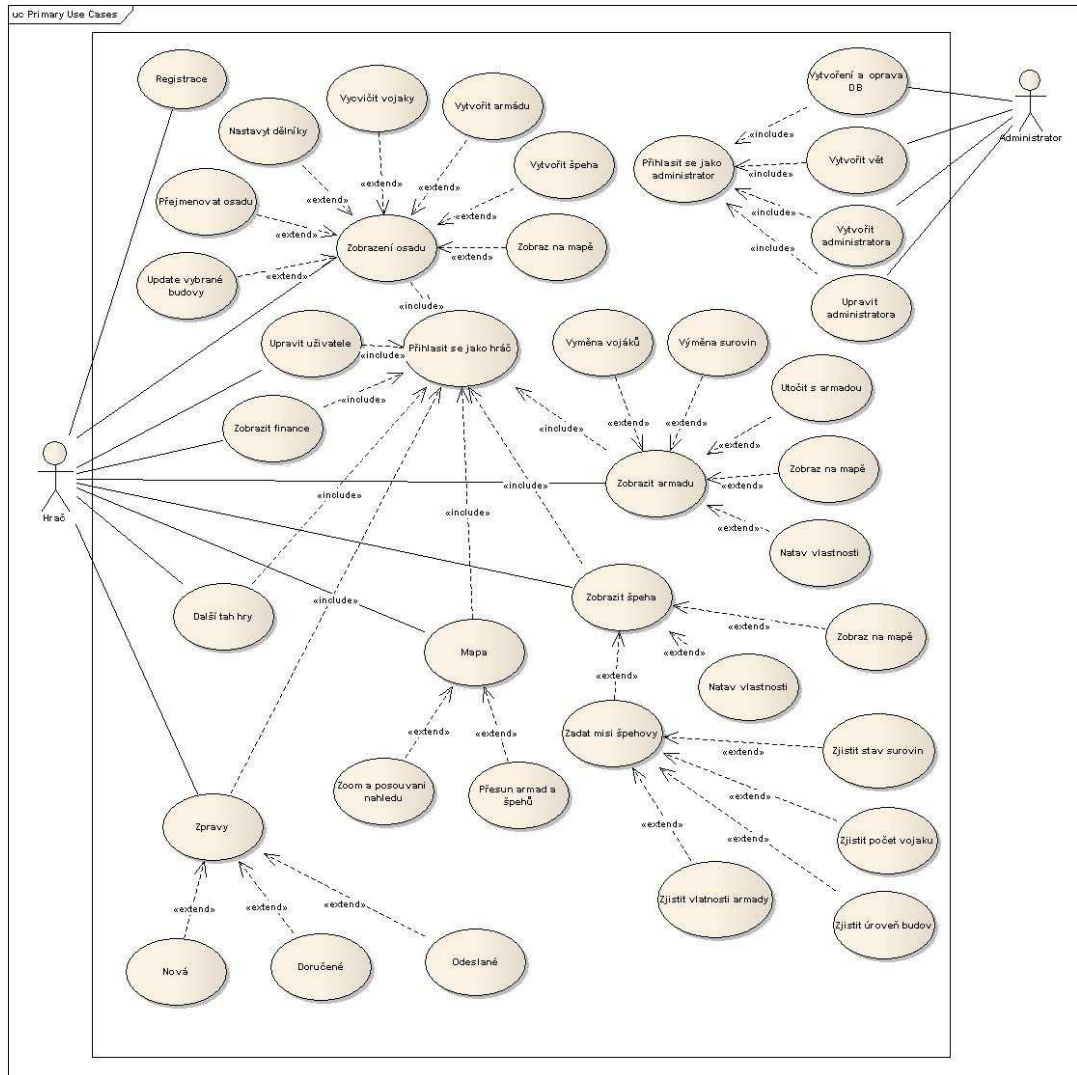


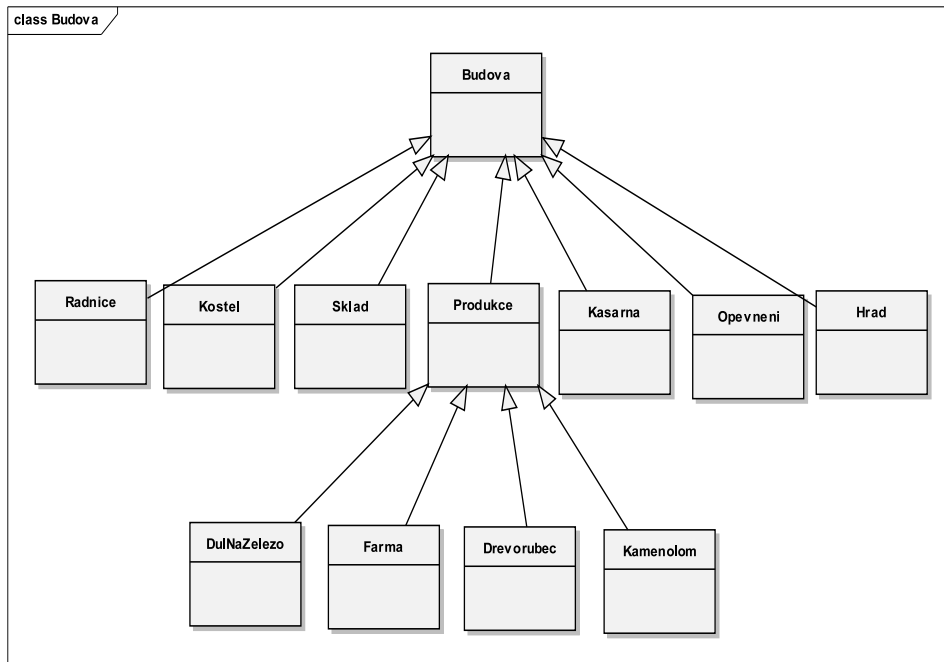
Diagram užití celého systému.

Na use case diagramu jsou znázorněny dvě role, hráč a administrátor. Systém administrátorovi příliš mnoho možností k řízení hry nenabízí, je však možnost hru editovat i přímým zásahem do databáze, je ale potřeba dodržet postupy popsané v sekci 5.. Hráč má v systému následující možnosti.

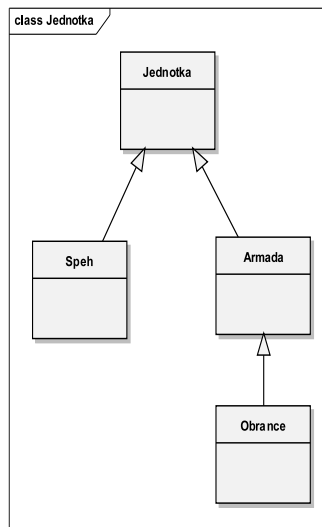
- Registrace: Hráč zadá jméno a heslo. Pokud ve hře neexistuje jiný hráč se stejným jménem a počet hráčů je menší než je maximální počet hráčů ve hře, vytvoří se nový hráčský účet.
- Upravit uživatele: Tato a následující možnosti použití systému, jsou pro hráče přístupné až po přihlášení do systému. Hráč si může změnit své přihlašovací údaje.
- Zobrazit finance: Slouží k zobrazení finančního přehledu hráče, kolik mu co vydělává a kolik ho co stojí.
- Zobrazení osady: Umožňuje zobrazit všechny osady hráče a pokud je to možné i editovat je.
- Zobrazit armádu: Umožňuje zobrazit všechny armády hráče a pokud je to možné i editovat je.
- Zobrazit špeha: Umožňuje zobrazit všechny špehy hráče a pokud je to možné i editovat je.
- Mapa: Zobrazuje jednotlivé objekty na mapě a objektům armáda a špeh umožňuje pohyb po mapě.
- Zprávy: Zde je možnost posílat a číst textové zprávy, zasílané mezi hráči.
- Další tah hry: Hráč má možnost povolit předčasné ukončení aktuálního tahu hry.

2.2. Class diagram

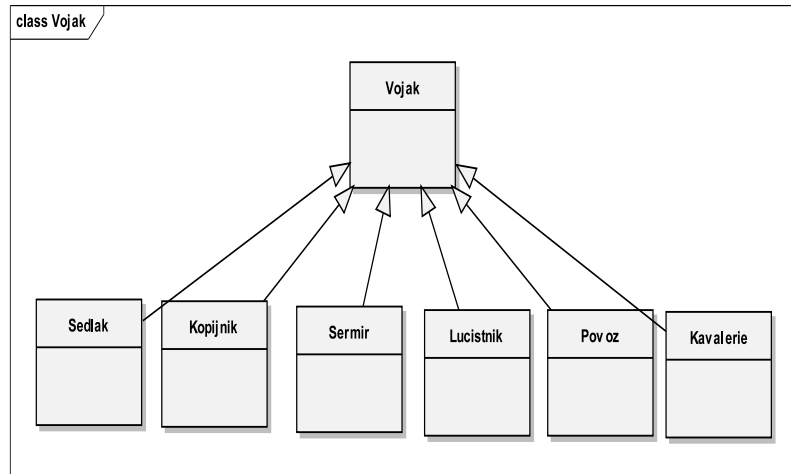
Diagram tříd zobrazím pro větší přehlednost na několika obrázcích.



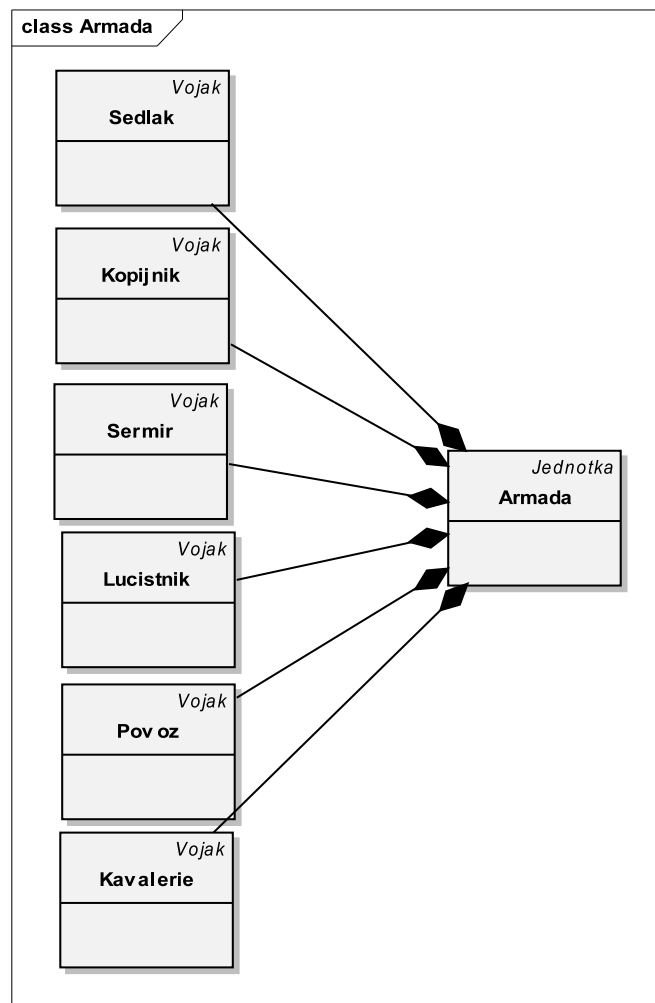
Na obrázku je znázorněna dědičnost všech budov vyskytujících se v systému.



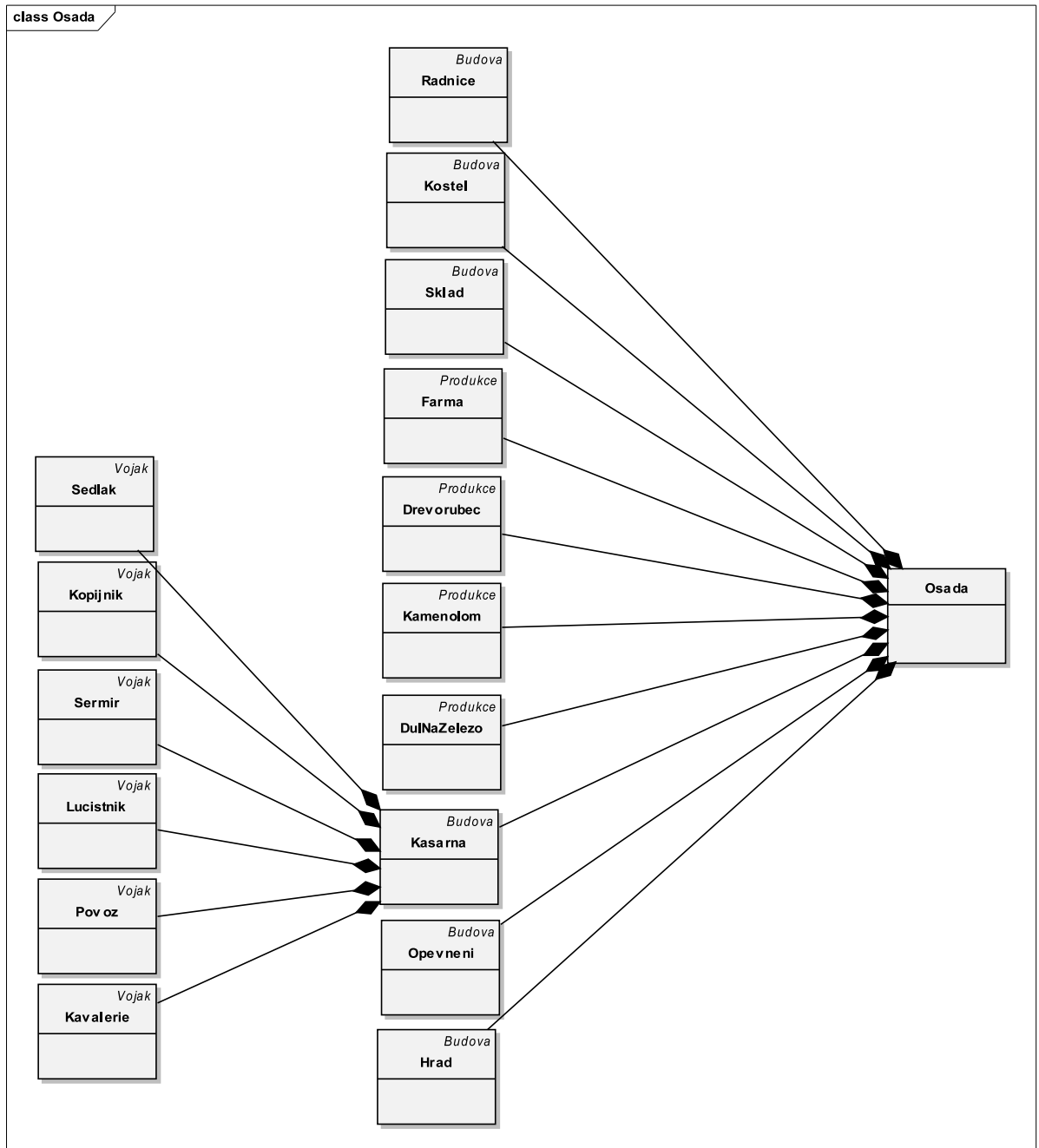
Obrázek znázorňuje dědičnost objektů „Speh“, „Armada“ a „Obrance“.



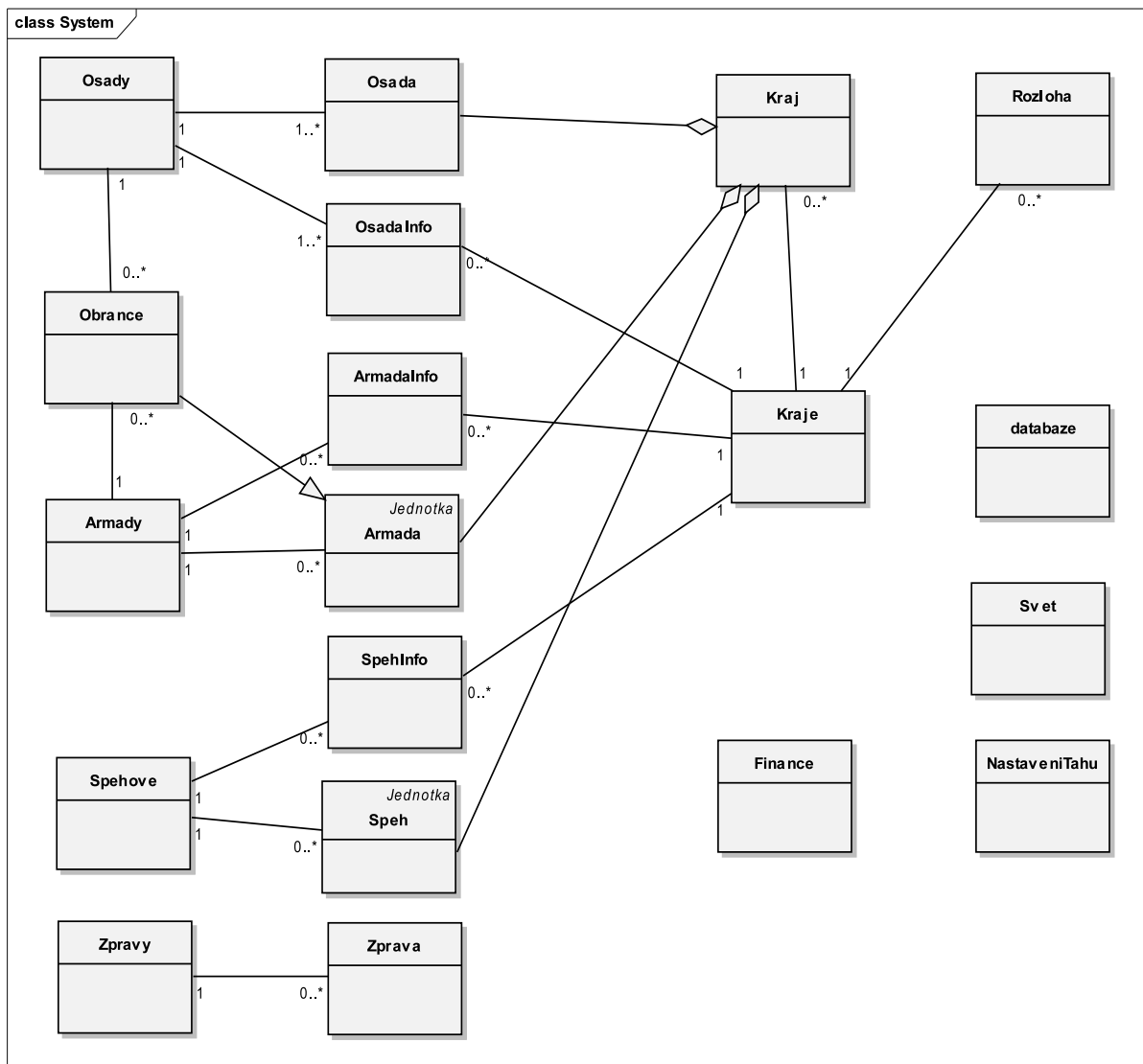
Na obrázku je znázorněna dědičnost všech vojenských jednotek vyskytujících se v systému.



Obrázek znázorňuje z jakých objektů se skládá objekt „Armada“.



Obrázek znázorňuje z jakých objektů se skládá objekt „Osada“.



Na obrázku jsou zobrazeny hlavní třídy, zajišťující převážnou většinu aplikační logiky.

Z předchozích obrázků je vše na první pohled hned jasné, výjimkou je však poslední obrázek a ten si nyní přiblížíme.

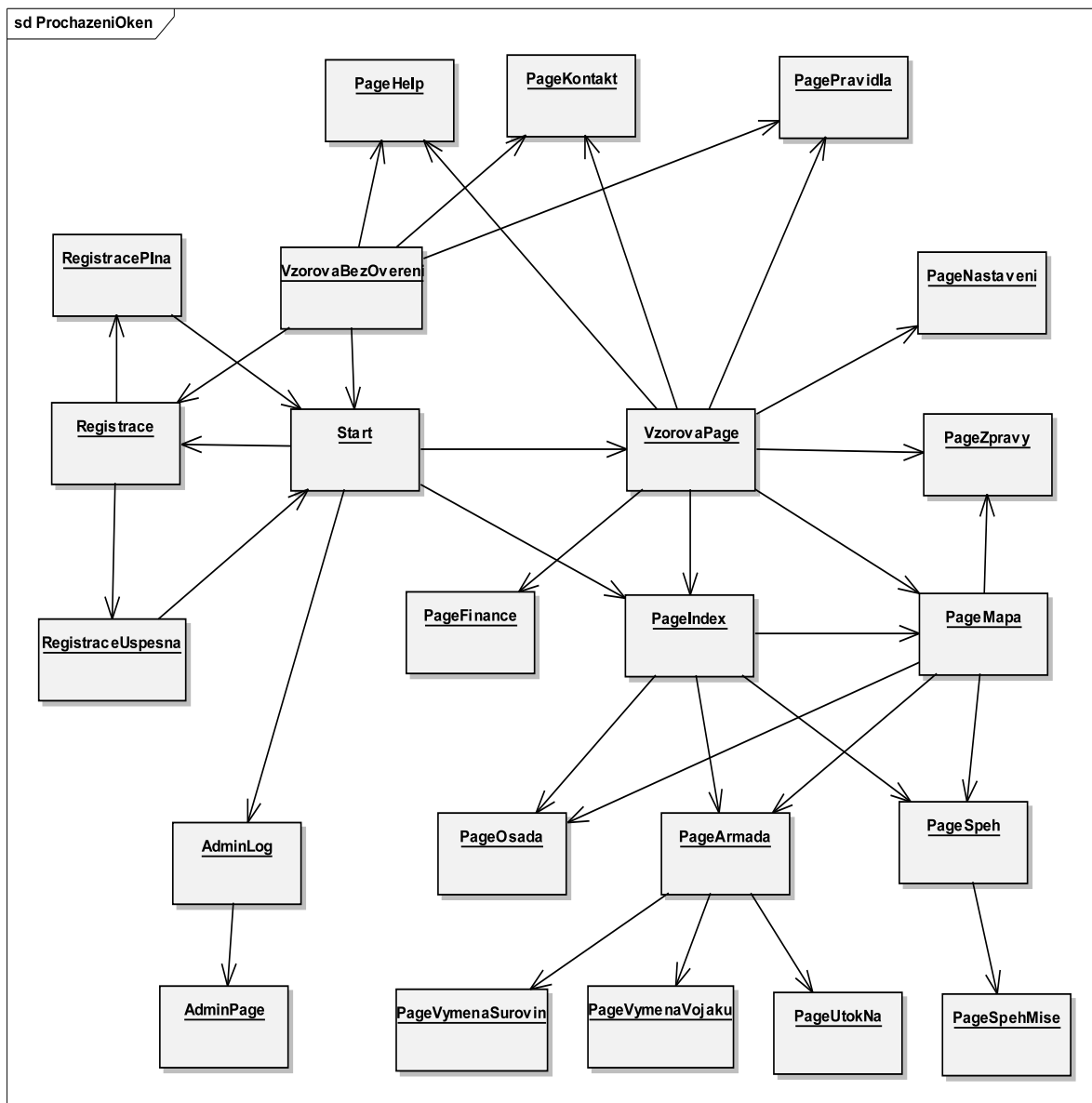
- Statická třída „Osady“: Tato třída obsahuje řadu metod, sloužících ke čtení a editaci objektu „Osada“ a objektu s osadou spojených (budovy a vojáci v osadě).
- Statická třída „Armady“: Tato třída obsahuje řadu metod, sloužících ke čtení a editaci objektu „Armada“ a objektu s armádou spojených (vojáci v armádě a osady napadené armádou).

- Statická třída „Spehove“: Tato třída obsahuje řadu metod, sloužících ke čtení a editaci objektu „Speh“.
- Statická třída „Kraje“: Tato třída obsahuje řadu metod, sloužících ke čtení a editaci objektu „Kraj“.
- Statická třída „Zpravy“: Tato třída obsahuje řadu metod, sloužících ke čtení a editaci objektu „Zprava“.
- Statická třída „Svet“: Tato třída obsahuje řadu metod, které přímo komunikují s databází a čtou či mění vlastnosti herního světa.

Úplný popis všech tříd a jejich funkcionalit najdete v sekci 4.

2.3. Uživatelské rozhraní

Webové stránky rozdělíme na stránky bez vzorové stránky, se vzorovou stránkou bez ověření a se vzorovou stránkou s ověřením. Jako první stránka se načítá stránka „Start“. K pozicování ovládacích prvků na webových stránkách používám kaskádové styly, umístěné v adresáři „App_Themes“.



Na obrázku je znázorněno procházení mezi webovými stránkami.

2.3.1. Bez vzorové stránky

- „AdminLog“: Tato stránka slouží pro přihlášení administrátora a není na ní žádný přímý odkaz z jiné stránky. Defaultní přihlašovací jméno a heslo je admin admin.
- „AdminPage“: Stránka slouží k základnímu nastavení a vytvoření celé hry. Stránka se může zobrazit ve čtyřech různých variantách v závislosti na hodnotě parametru „volba“.

- Založení a oprava databázového systému: Slouží k založení databázového systému, opravě procedur, pohledů a triggerů.
- Možnosti k vytváření a úpravě hry: Slouží k vytvoření nové hry.
- Vytvoření administrátorského účtu: Zde se dá vytvořit nový administrátorský účet.
- Úprava administrátorského účtu: Slouží k úpravě právě přihlášeného administrátorského účtu.

2.3.2. S vzorovou stránkou bez ověření

- „VzorovaBezOvereni“: Nejedná se o klasickou webovou stránku, ale o vzor (společné vlastnosti) pro ostatní stránky, který ostatní stránky dědí. Z tohoto vzoru je pak odkaz na stránky „Start“, „Registrace“, „PageHelp“, „PageKontakt“ a „PagePravidla“.
- „Start“: Tato stránka se načítá jako první. Dědí ze stránky „VzorovaBezOvereni“. Nabízí přihlášení hráče do hry nebo přechod na stránku pro zaregistrování nového hráče.
- „Registrace“: Dědí ze stránky „VzorovaBezOvereni“. Umožňuje zaregistrování nových hráčů (pokud to systém ještě dovolí).
- „RegistracePlna“: Dědí ze stránky „VzorovaBezOvereni“. Tato stránka se načte, pokud je v systému již registrováno maximum hráčů a někdo další se pokusí zaregistrovat. Informuje tedy o nemožnosti registrace z důvodu plného obsazení jinými hráči.
- „RegistraceUspesna“: Dědí ze stránky „VzorovaBezOvereni“. Pokud je v systému ještě volno, informuje právě zaregistrovaného hráče o úspěšné registraci.

2.3.3. S vzorovou stránkou s ověřením

- „VzorovaPage“: Nejedná se o klasickou webovou stránku, ale o vzor (společné vlastnosti) pro ostatní stránky, který ostatní stránky dědí. Z tohoto vzoru je pak odkaz na stránky „PageFinance“, „PageIndex“, „PageMapa“, „PageZpravy“, „PageNastaveni“, „PageHelp“, „PageKontakt“ a „PagePravidla“.
- „PageIndex“: Dědí ze stránky „VzorovaPage“. Tato stránka se objeví jako první po přihlášení hráče. Stránka slouží k zobrazení seznamů všech osad, armád a špehů právě přihlášeného hráče. Stránka se může zobrazit ve třech různých variantách v závislosti parametru „typObjektu“, každá z těchto

variant zobrazuje jiný seznam objektů. V seznamu lze kliknout na název objektu, čímž se přejde na stránku zobrazující vybraný objekt nebo můžete kliknout na číslo pozice daného objektu a tím se načte stránka zobrazující mapu, v jejímž středu se bude nacházet právě vybraný objekt.

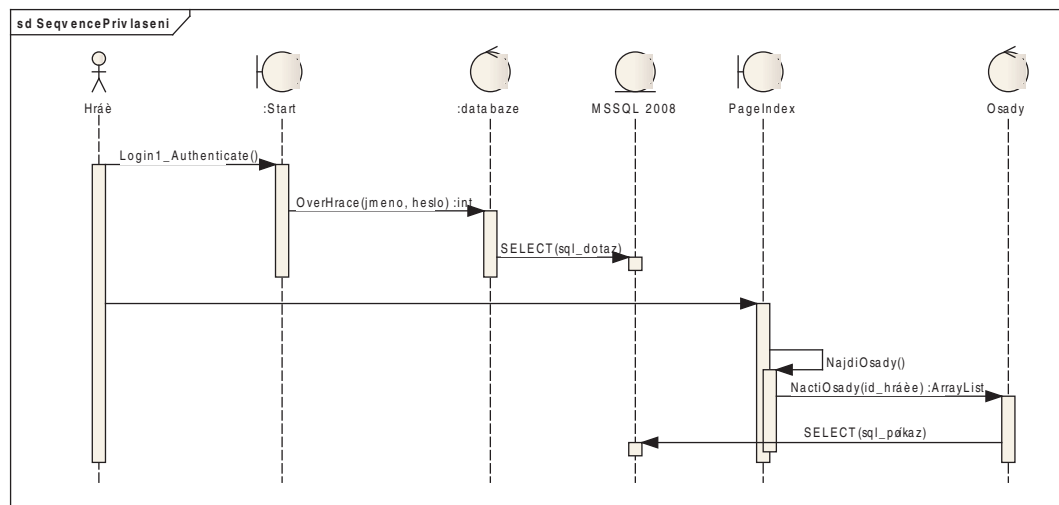
- bez parametru: Pokud parametr není zadán, načte se seznam všech osad právě přihlášeného hráče.
 - parametr „armada“: Pokud hodnota parametru je rovna „armada“, načte se seznam všech armád právě přihlášeného hráče.
 - parametr „speh“: Pokud hodnota parametru je rovna „speh“, načte se seznam všech špehů právě přihlášeného hráče.
- „PageFinance“: Dědí ze stránky „VzorovaPage“. Zobrazuje přehled financování všech objektů právě přihlášeného hráče.
 - „PageMapa“: Dědí ze stránky „VzorovaPage“. Zobrazuje herní mapu světa. Na této mapě jsou znázorněny všechny objekty, kromě špehů nepatřících právě přihlášenému hráči. Stránka přebírá několik parametrů, určujících polohu na mapě, zoom a zda je či není označen některý z objektů.
 - bez parametru: Načte se mapa s největším zoom a v jejímž středu je hlavní (první) osada právě přihlášeného hráče.
 - parametr „pozice“: Tento parametr určuje index kraje, který má být ve středu zobrazené mapy.
 - parametr „dim“: Tento parametr určuje počet krajů na čtvercové straně zobrazené mapy.
 - parametr „objekt“: Tento parametr určuje jaký objekt je vybrán. Hodnoty tohoto parametru mohou být:
 - * jen číslo: Číslo určuje index vybrané osady.
 - * písmeno „a“ a za ním číslo: Číslo určuje index vybrané armády.
 - * písmeno „s“ a za ním číslo: Číslo určuje index vybraného špeha.
 - „PageZpravy“: Dědí ze stránky „VzorovaPage“. Zobrazuje doručené a odeslané zprávy a umožňuje je i psát.
 - „PageNastaveni“: Dědí ze stránky „VzorovaPage“. Umožňuje změnu přihlašovacích údajů.
 - „PageOsada“: Dědí ze stránky „VzorovaPage“. Zobrazuje jednotlivé budovy osady. Na tuto stránku se dá dostat buď ze seznamu osad nebo z mapy. V levém menu je seznam budov, uprostřed pak co daná budova umožňuje a v pravém sloupci je možnost zvýšení či snížení úrovně budovy a přehled

skladových zásob surovin. Která budova se má zobrazit je dáno parametrem „Budova“.

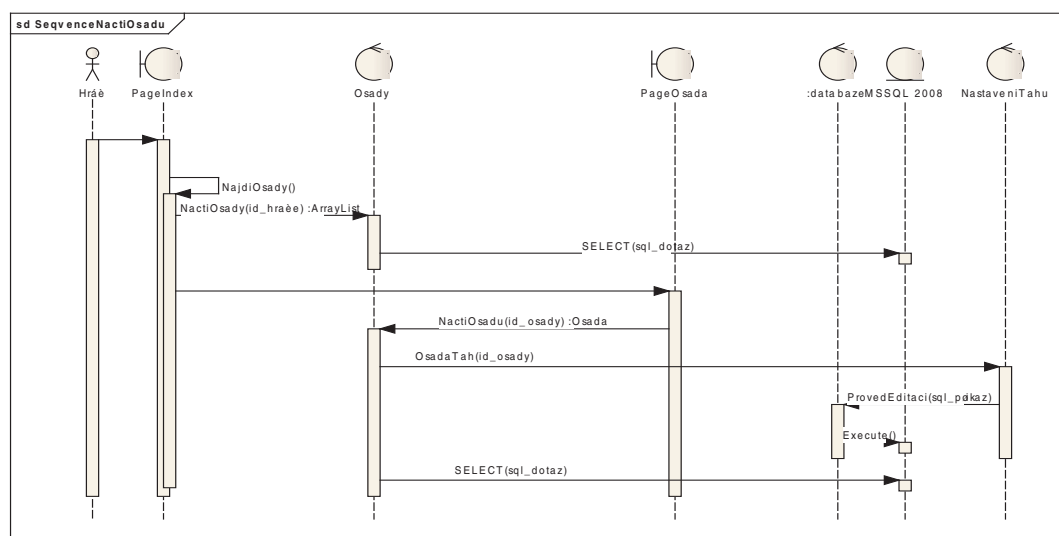
- parametr roven „1“ nebo bez parametru: Načte se základní přehled osady a budova radnice.
 - parametr roven „2“: Načte se budova kostel.
 - parametr roven „3“: Načte se budova sklad.
 - parametr roven „4“: Načte se budova farma.
 - parametr roven „5“: Načte se budova dřevorubec.
 - parametr roven „6“: Načte se budova kamenolom.
 - parametr roven „7“: Načte se budova důl na železo.
 - parametr roven „8“: Načte se budova kasárna.
 - parametr roven „9“: Načte se budova hrad.
 - parametr roven „10“: Načte se budova opevnění.
- „PageArmada“: Dědí ze stránky „VzorovaPage“. Zobrazuje možnosti armády. Na tuto stránku se dá dostat buď ze seznamu armád, nebo z mapy.
 - „PageSpeh“: Dědí ze stránky „VzorovaPage“. Zobrazuje možnosti špeha. Na tuto stránku se dá dostat buď ze seznamu špehů, nebo z mapy.
 - „PageVymenaSurovin“: Dědí ze stránky „VzorovaPage“. Na tuto stránku se dá dostat ze stránky „PageArmada“ a umožňuje výměnu surovin mezi armádou a armádou nebo mezi armádou a osadou. Pokud oba objekty, mezi kterými má výměna nastat, patří jednomu hráči, je možná oboustranná výměna, jinak je možný jen expotr. Výměna však může nastat, jen pokud oba objekty jsou na stejné pozici.
 - „PageVymenaVojaku“: Dědí ze stránky „VzorovaPage“. Na tuto stránku se dá dostat ze stránky „PageArmada“ a umožňuje výměnu vojáků mezi armádou a armádou nebo mezi armádou a osadou. Oba objekty, mezi kterými má výměna nastat, musí patřit jednomu hráči a musí se nacházet na stejné pozici.
 - „PageUtokNa“: Dědí ze stránky „VzorovaPage“. Na tuto stránku se dá dostat ze stránky „PageArmada“ a provádí útok armády na armádu nebo na osadu. Oba objekty, mezi kterými má bitva proběhnout musí být na stejné pozici.
 - „PageMiseSpeh“: Dědí ze stránky „VzorovaPage“. Na tuto stránku se dá dostat ze stránky „PageSpeh“ a provádí špionáž armády nebo osady. Objekt určený ke špionáži musí být na stejné pozici jako špeh.

2.4. Sekvenční diagramy

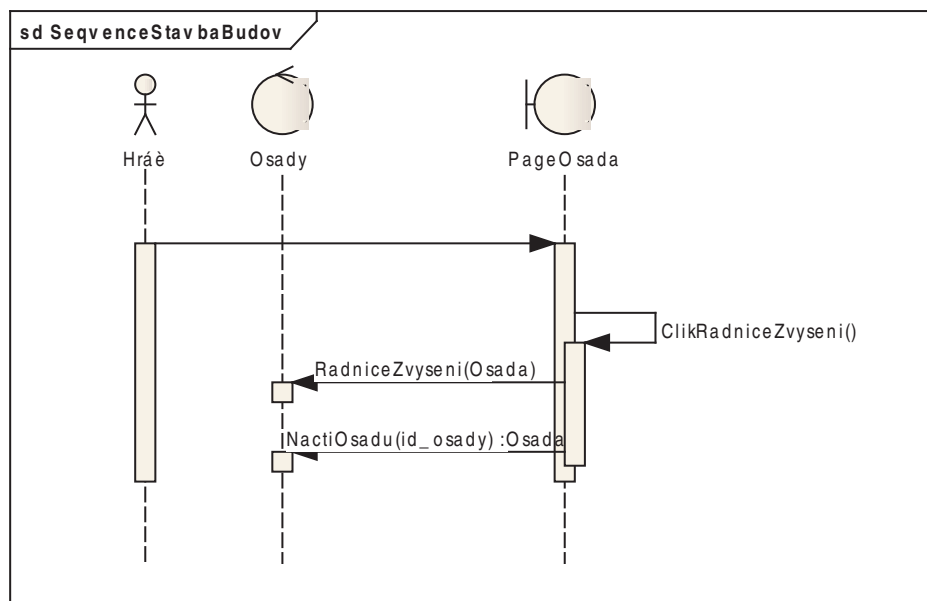
Níže zobrazené sekvenční diagramy neznázorňují zdaleka všechny možné posloupnosti, které v systému mohou nastat. Vybral jsem ale takové, podle kterých, by neměl být problém odvodit si zbývající.



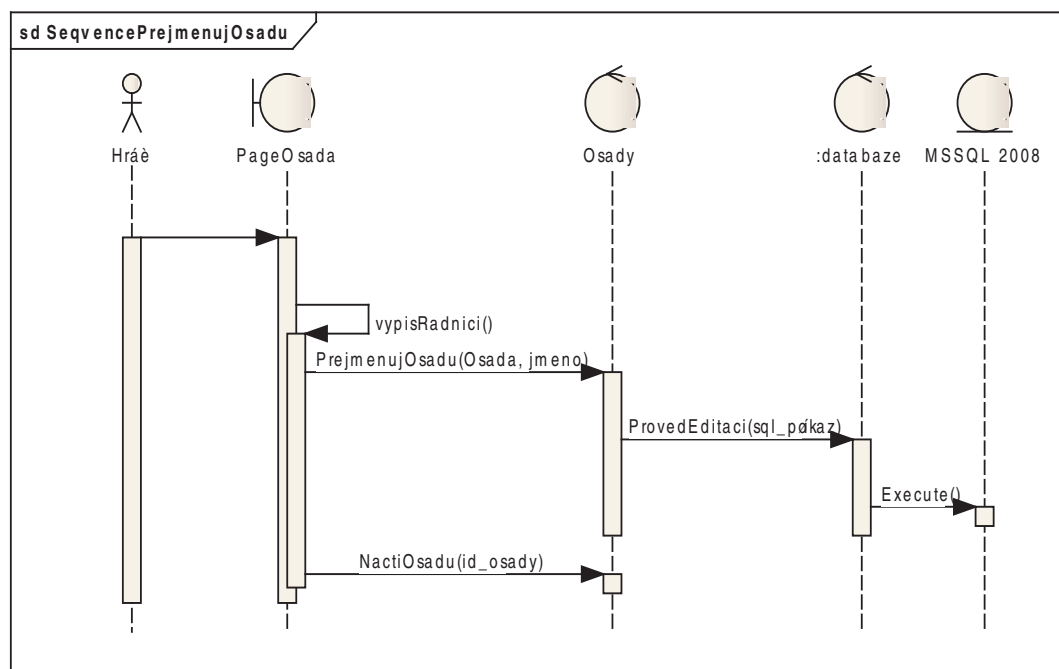
Posloupnost vykonávání metod, při přihlášení hráče.



Posloupnost vykonávání metod, při zobrazení osady. V této posloupnosti je vynechaná část přihlášení, ta je již popsána.

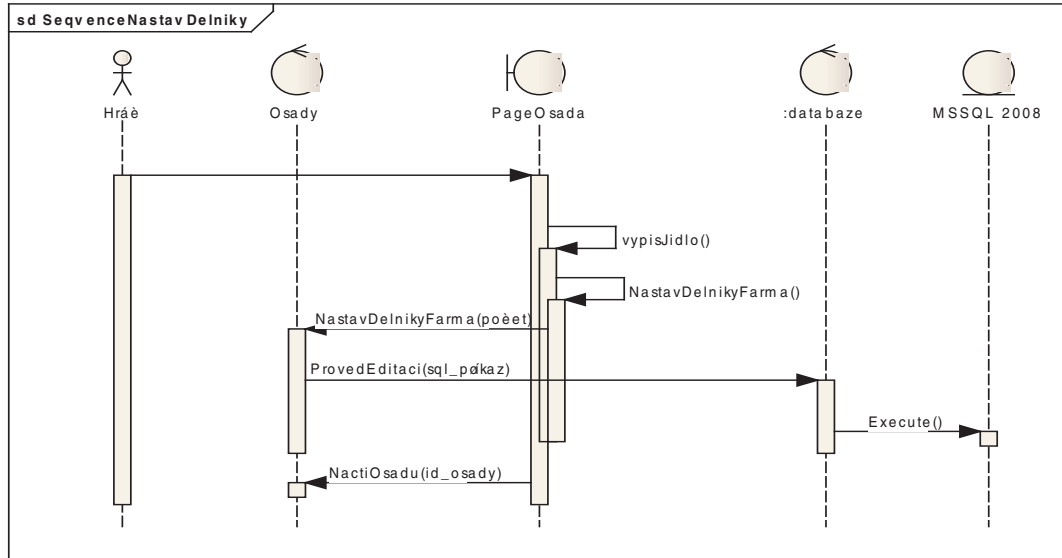


Posloupnost metod pro výstavbu budovy radnice. Tato posloupnost vychází z posloupnosti načtení osady a rozšiřuje ji. Podobně vypadá i posloupnost pro bourání radnice, akorát místo volání metody „RadniceZvyseni“ se volá metoda „RadniceSnizeni“. Obdobně vypadají i posloupnosti pro zvýšení, či snížení úrovně ostatních budov v osadě.

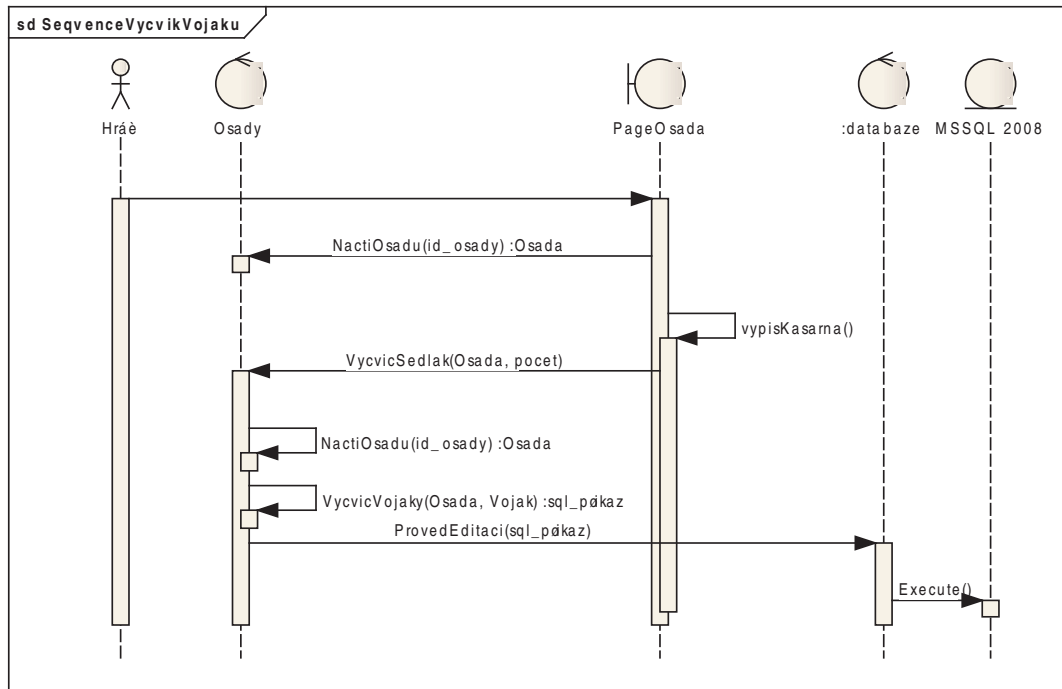


Posloupnost metod pro přejmenování osady. Tato posloupnost vychází z posloupnosti načtení osady a rozšiřuje ji. Obdobně by vypadala i posloupnost

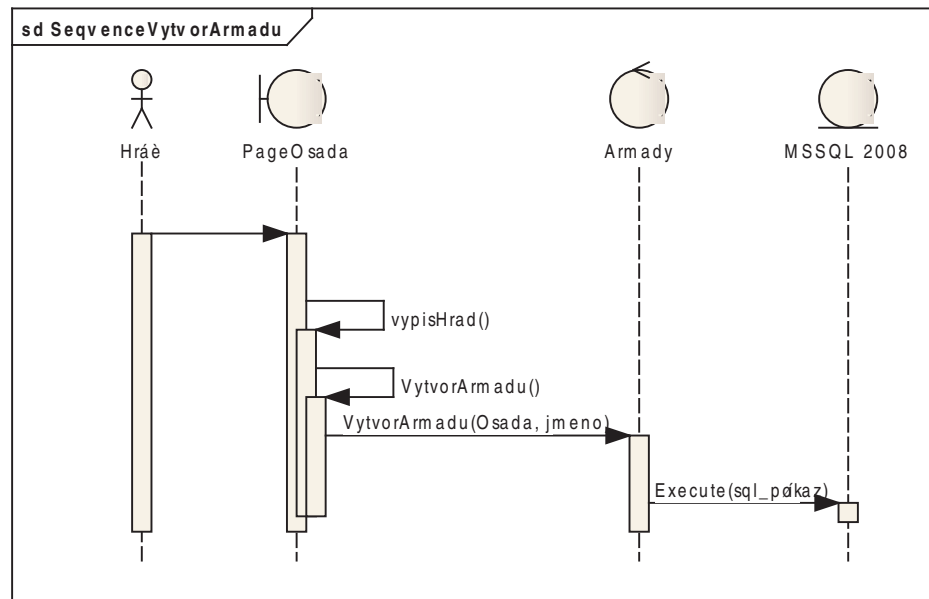
metod pro přejmenování armády, či špeha.



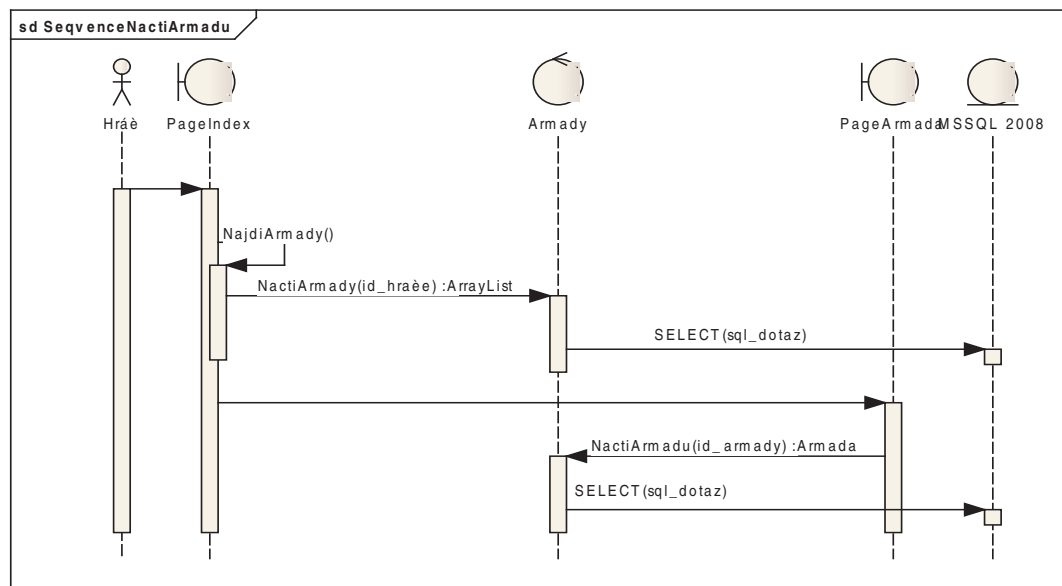
Posloupnost metod pro nastavení počtu dělníků pracujících na farmě. Tato posloupnost vychází z posloupnosti načtení osady a rozšiřuje ji. Obdobně by vypadala i posloupnost metod nastavení počtu pracujících dělníků v ostatních produkcích.



Posloupnost metod pro výcvik sedláků v kasárnách. Tato posloupnost vychází z posloupnosti načtení osady a rozšiřuje ji. Obdobně by vypadala i posloupnost metod výcviku ostatních druhů jednotek.

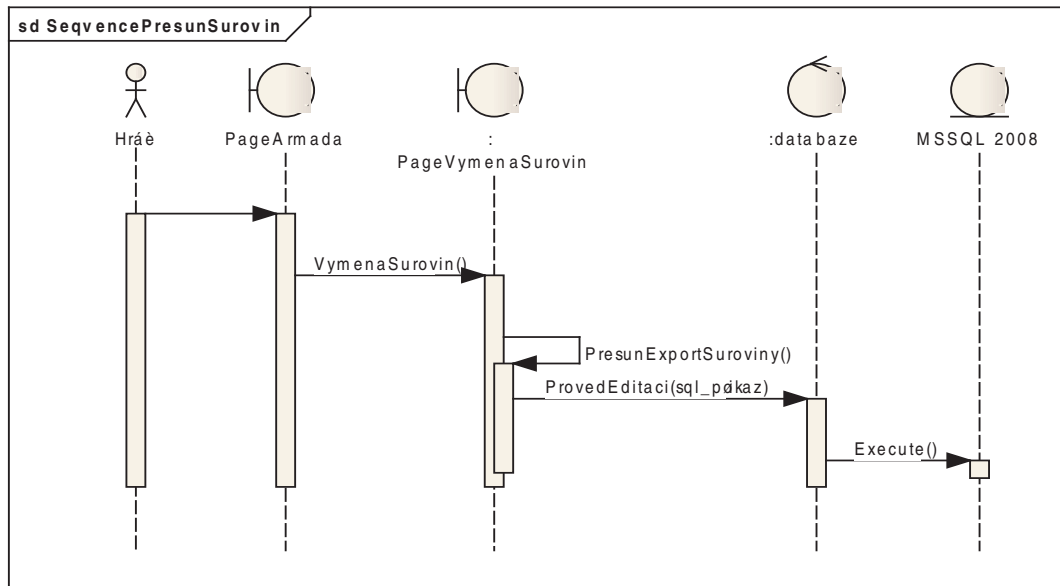


Posloupnost metod při vytváření armády. Tato posloupnost vychází z posloupnosti načtení osady a rozšiřuje ji. Obdobně by vypadala i posloupnost metod při vytváření špeha.

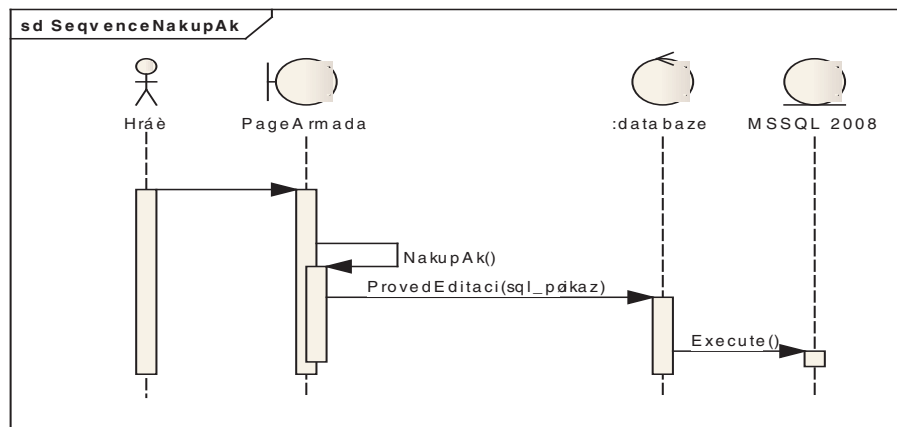


Posloupnost vykonávání metod, při zobrazení armády. V této posloupnosti je vynechaná část přihlášení, ta je již popsána. Obdobným způsobem by vypadala

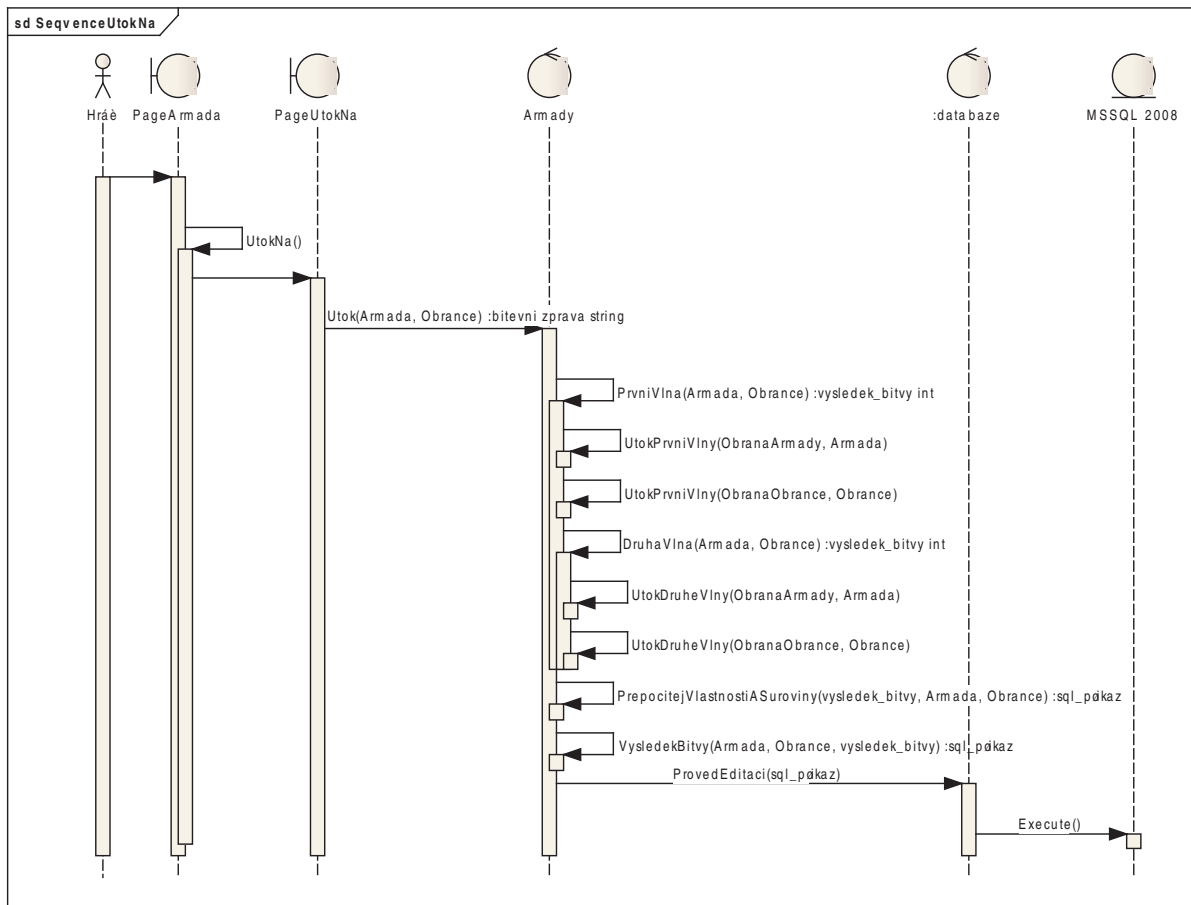
i posloupnost metod pro zobrazení špeha.



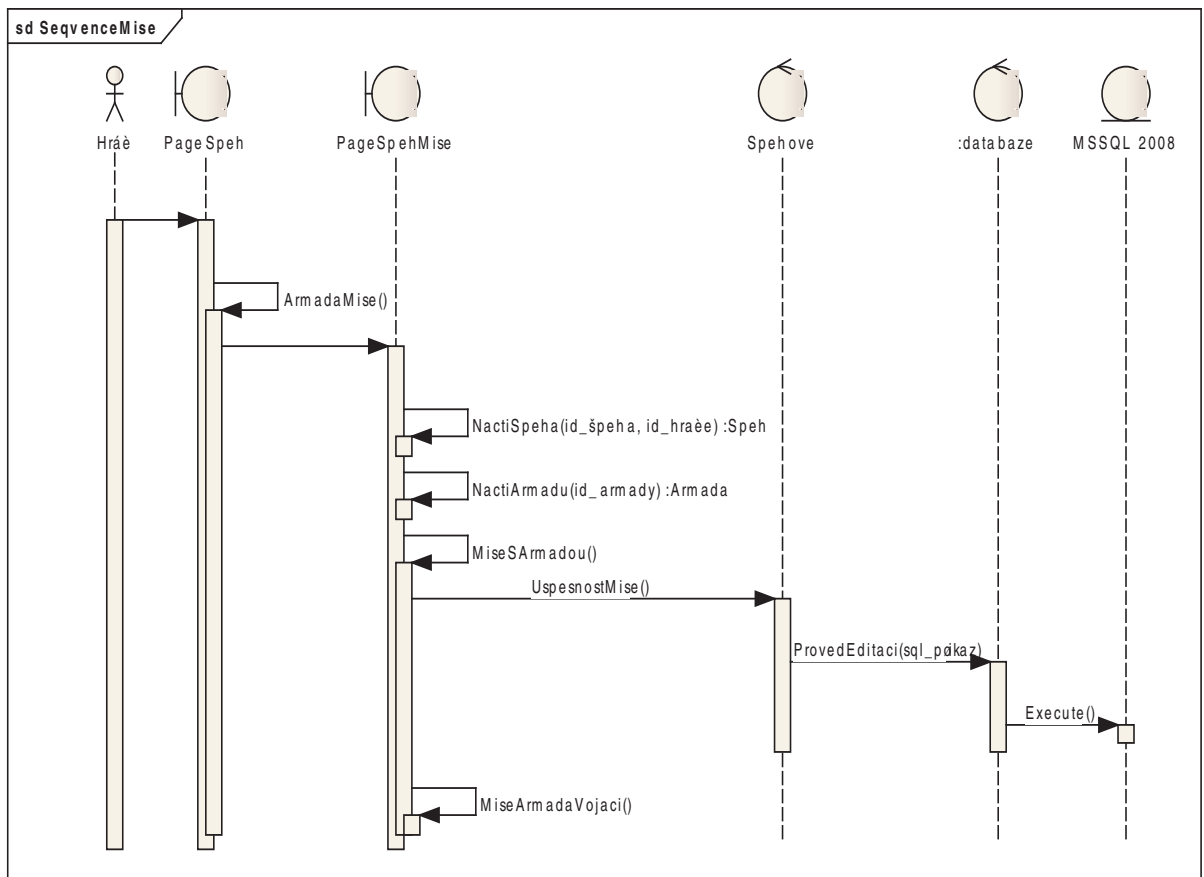
Posloupnost metod při exportu surovin mezi objekty. Tato posloupnost vychází z posloupnosti načtení armády a rozšiřuje ji. Obdobným způsobem by vypadala i posloupnost metod pro import surovin a přesun vojáků.



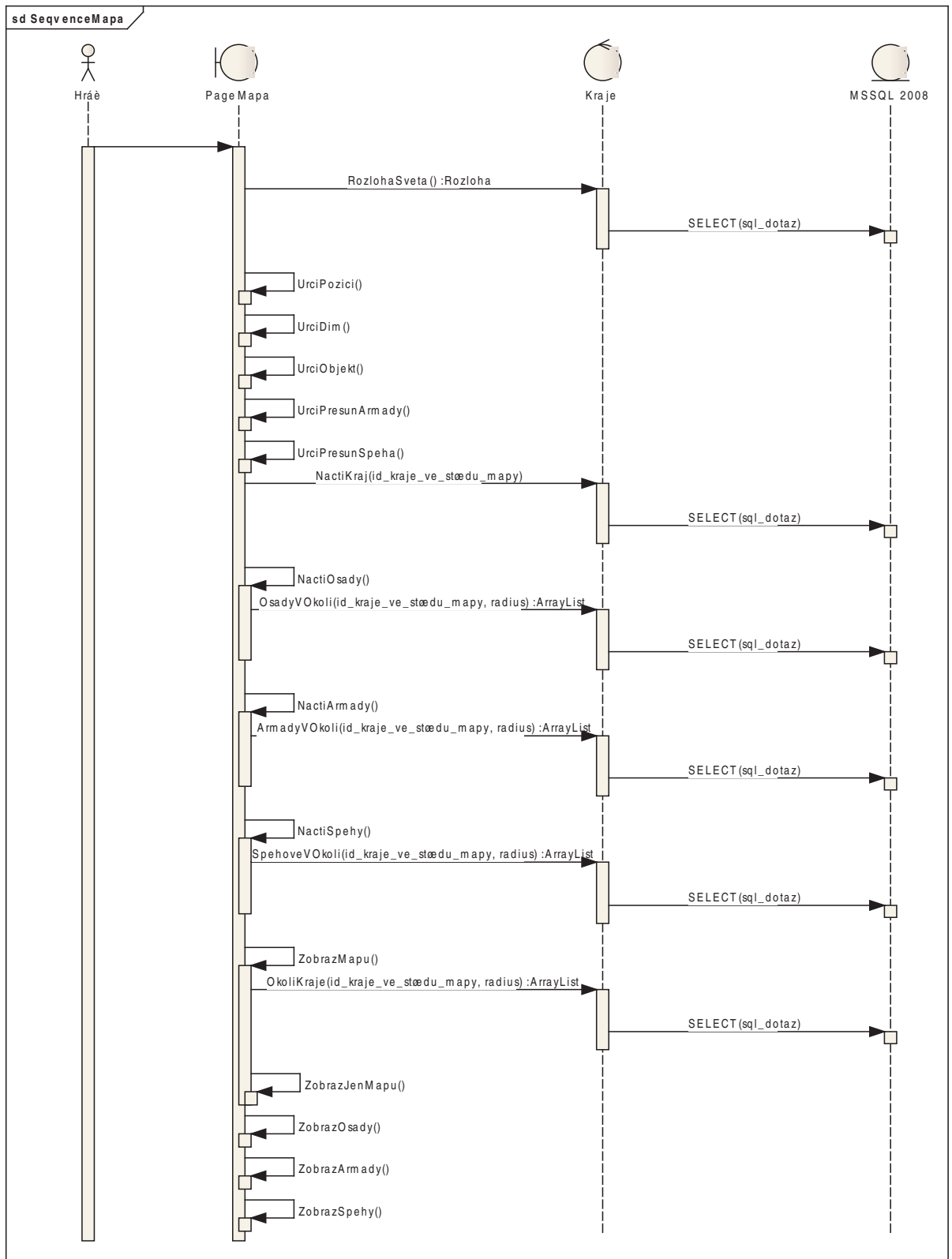
Posloupnost metod při nákupu akčních bodů v armádě. Tato posloupnost vychází z posloupnosti načtení armády a rozšiřuje ji. Obdobným způsobem by vypadala posloupnost nákupu morálky v armádě, nákupu akčních bodů a obratnosti u špeha.



Posloupnost metod při útoku armády. Tato posloupnost vychází z posloupnosti načtení armády a rozšiřuje ji. Je to konkrétní případ pro bitvy trvající obě bitevní kola. Může nastat i zkrácená situace, kdy k druhému kolu již nedochází.



Posloupnost metod při špehem prováděné misi, pro zjištění počtu vojáků v cizí armádě. Tato posloupnost vychází z posloupnosti načtení špeha a rozšiřuje ji. Obdobným způsobem by vypadaly i posloupnosti dalších misí, které může špeh provádět.



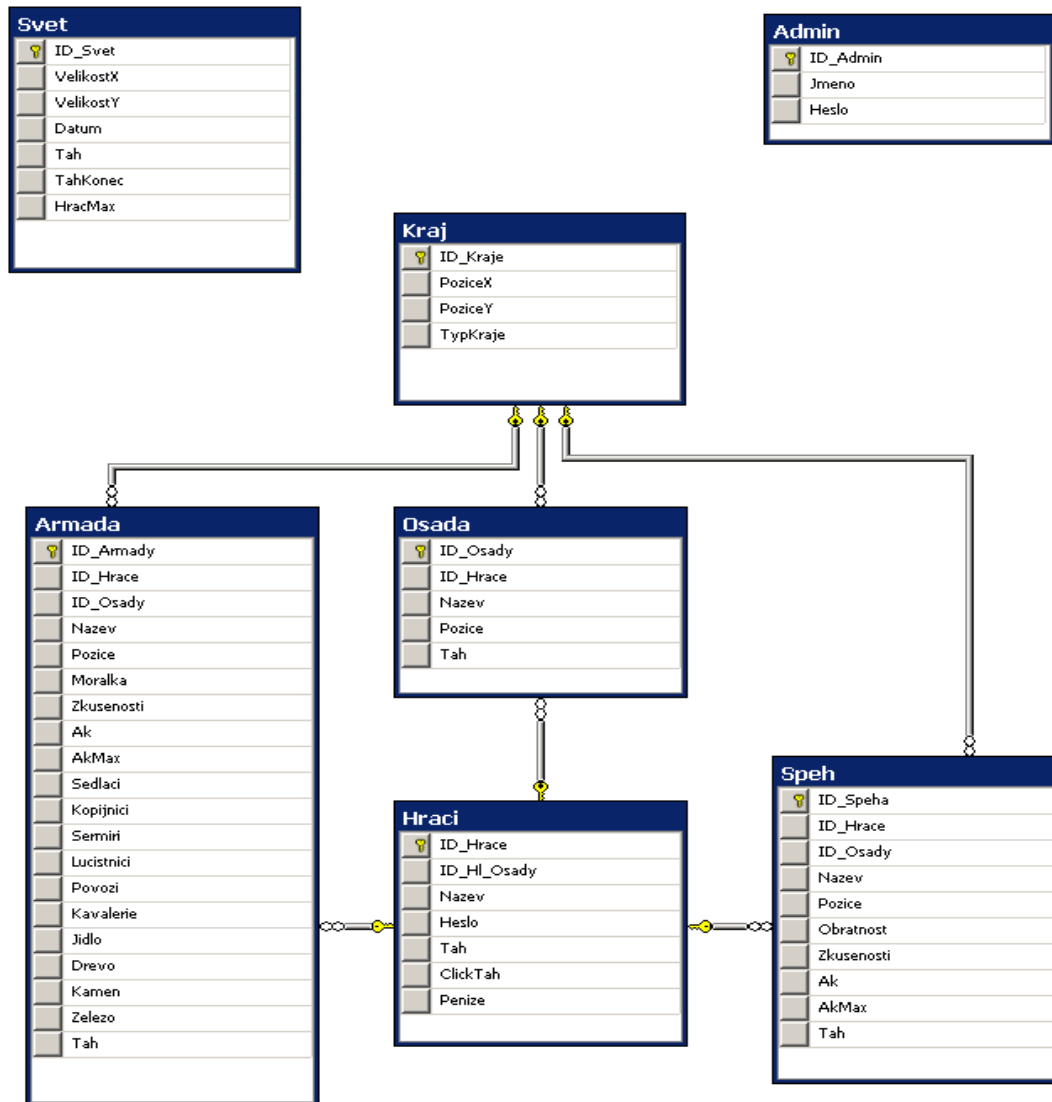
Posloupnost metod zobrazení mapy. V této posloupnosti je vynechaná část

přihlášení, ta je již popsána. Posloupností pro zobrazení je zase více, tato nám zobrazí jen mapu, bez výběru nějakého objektu. Další možnosti zobrazení mapy jsou podobné.

3. Databáze

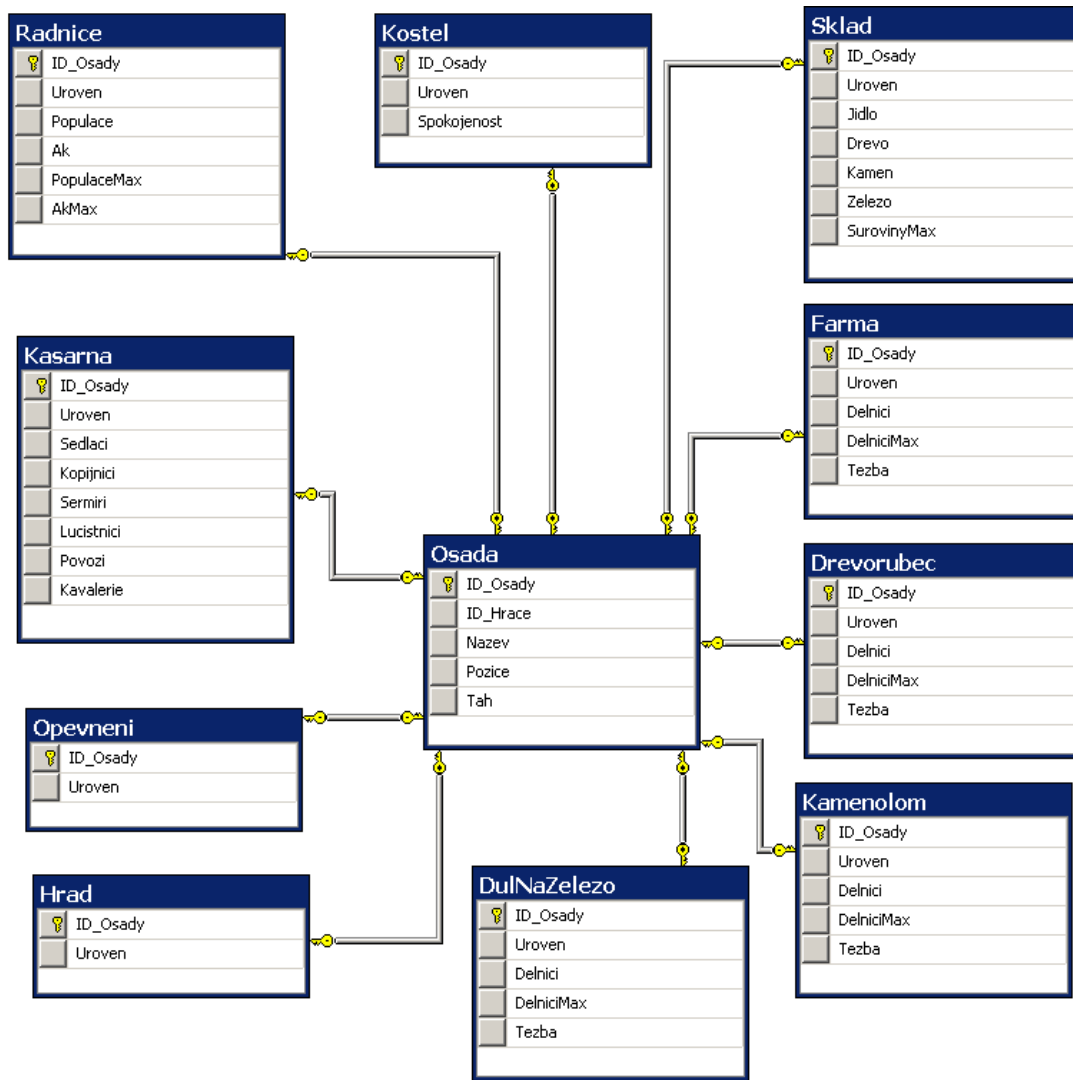
Databáze se skládá z tabulek, triggerů, pohledů a uložených procedur. Pro nahlédnutí jsou v adresáři „Databaze/SQL“ uloženy SQL scripty pro vytvoření celého databázového systému.

3.1. Tabulky



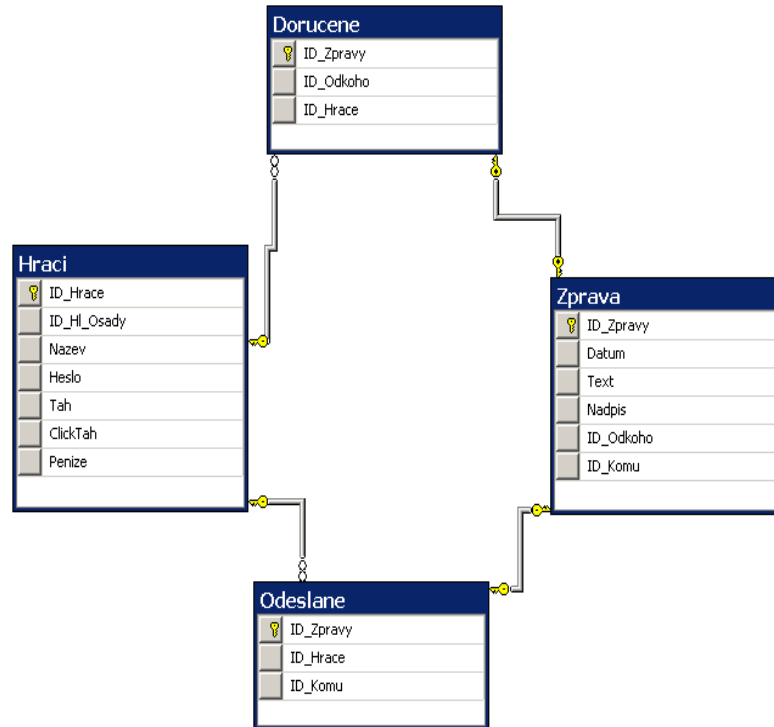
Hlavní systém tabulek.

- Popis: Tento systém tabulek znázorňuje závislost tabulek „Armada“, „Speh“ a „Osada“ na tabulce „Kraj“ a „Hrac“.



Tabulky potřebné k vytvoření osady.

- Popis: Tento systém tabulek znázorňuje závislost tabulek představujících jednotlivé budovy, na tabulce „Osada“.



Tabulky potřebné k posílání zpráv.

- Popis: Tento systém tabulek zajišťuje bezpečné zasílání zpráv mezi hráči. Každá zpráva, má kromě samotného textu, i informace o odesílateli a příjemci. Pokud odesílatel (příjemce) zadá zprávu smazat, smaže se záznam z tabulky „Odeslane“ („Dorucene“) a z tabulky „Zprava“ se smaže jen, pokud v tabulce „Dorucene“ („Odeslane“) záznam o této zprávě není. To zaručí, aby byla zpráva pro příjemce (odesílatele) dostupná, i když odesílatel (příjemce) dal příkaz ke smazání zprávy.

3.2. Triggery

3.2.1. KontrolaDrevorubec

- Popis: Při změně údajů v tabulce „Drevorubec“, nedovolí nastavit více dělníků, než je limit budovy a méně než nula.

3.2.2. KontrolaDulNaZezezo

- Popis: Při změně údajů v tabulce „DulNaZezezo“, nedovolí nastavit více dělníků, než je limit budovy a méně než nula.

3.2.3. KontrolaFarma

- Popis: Při změně údajů v tabulce „Farma“, nedovolí nastavit více dělníků, než je limit budovy a méně než nula.

3.2.4. KontrolaKamenolom

- Popis: Při změně údajů v tabulce „Kamenolom“, nedovolí nastavit více dělníků, než je limit budovy a méně než nula.

3.2.5. KontrolaSklad

- Popis: Při změně údajů v tabulce „Sklad“, nedovolí nastavit více surovin, než dovoluje limit budovy.

3.2.6. KontrolaSklad

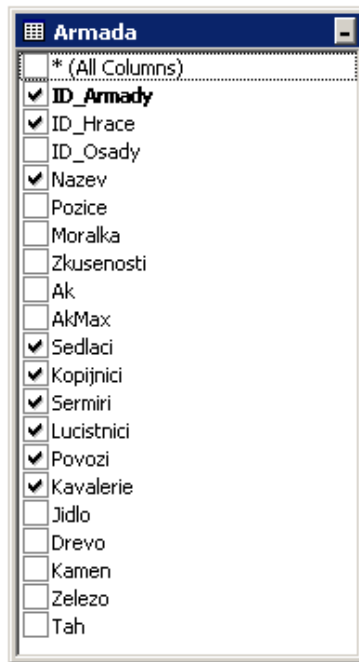
- Popis: Při vložení nového hráče do tabulky „Hraci“, nastaví atribut „Tah“ na aktuální tah hry a atributy „ClikTah“ a „Penize“ na nula.

3.2.7. VlozeniZpravy

- Popis: Při vložení nové zprávy do tabulky „Zprava“, vloží i patřičné údaje do tabulek „Odeslane“ a „Dorucene“.

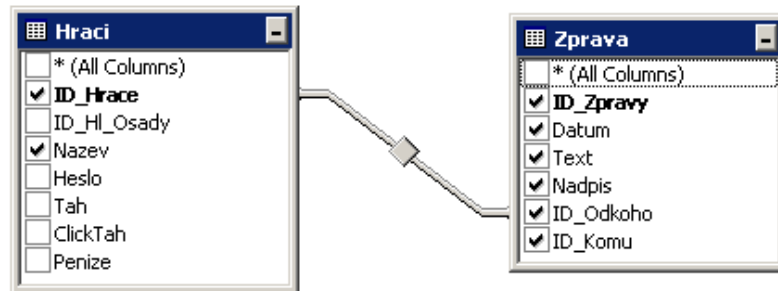
3.3. Pohledy

3.3.1. ViewArmadaFinance



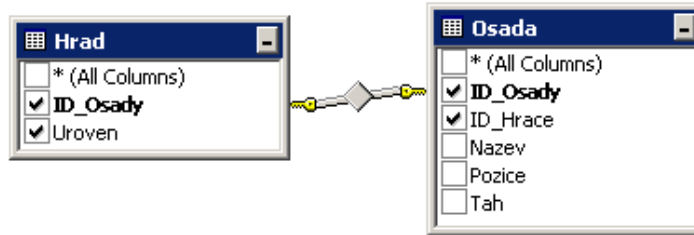
Tento pohled vybírá z tabulky „Armada“ jen data, potřebná k výpočtu všech nákladů na armádu.

3.3.2. ViewDorucene



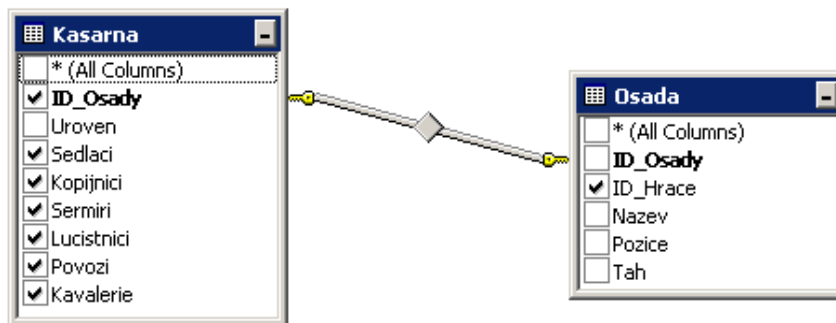
Tento pohled vybírá z tabulek „Hrac“ a „Zprava“ data, potřebná k zobrazení doručené zprávy.

3.3.3. View_Hrady



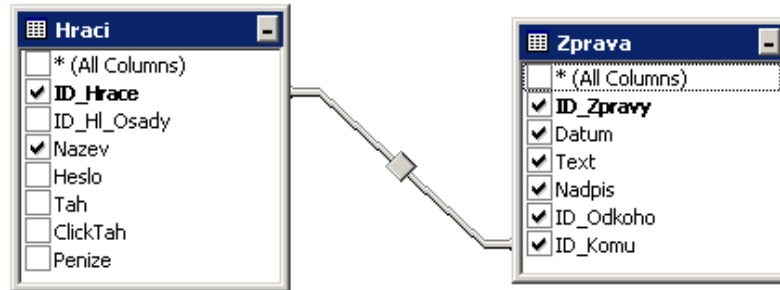
Pohled načítá z tabulek „Osada“ a „Hrad“ data, potřebná k určení počtu hradů jednoho hráče. Na základě tohoto počtu se pak stanovuje limit počtu armád a špehů.

3.3.4. ViewKasarnaVlastnik



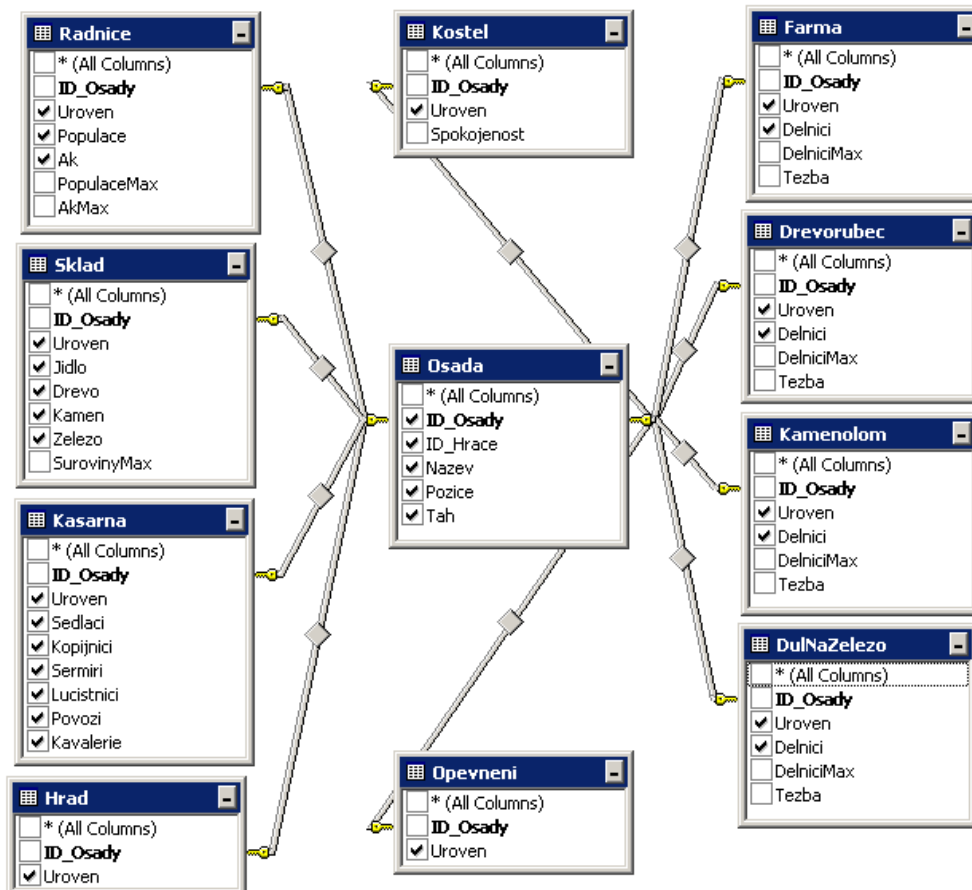
Tento pohled vybírá z tabulek „Kasarna“ a „Osada“ data, potřebná k určení, kterému hráči patří vojáci v kasárnách.

3.3.5. ViewOdeslane



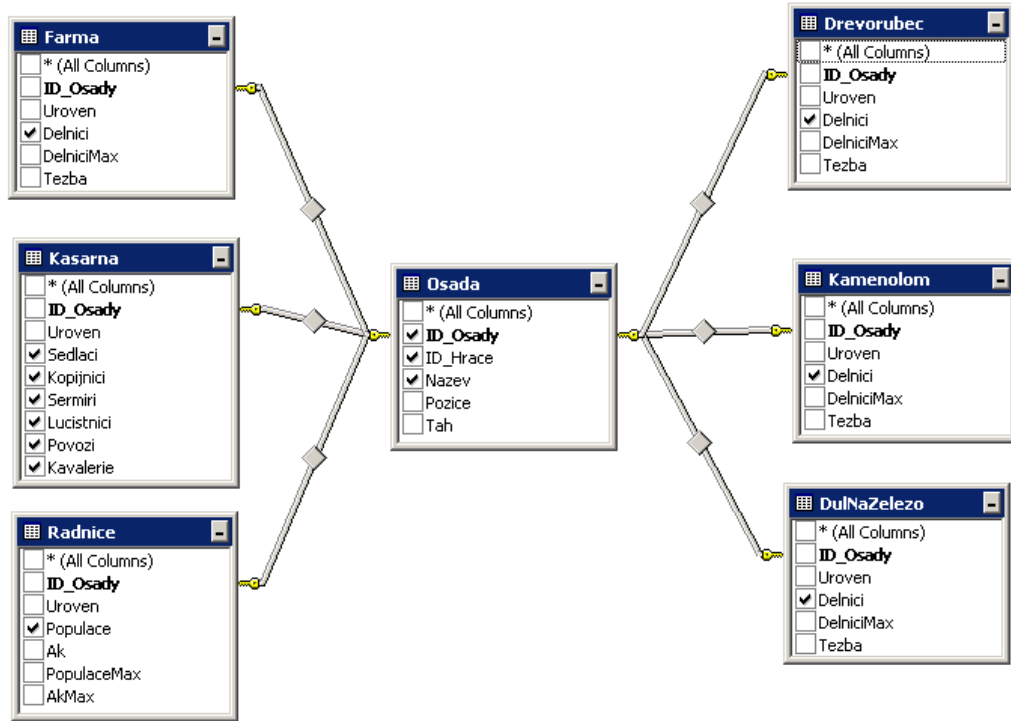
Tento pohled vybírá z tabulek „Hrac“ a „Zprava“ data, potřebná k zobrazení odeslané zprávy.

3.3.6. ViewOsada



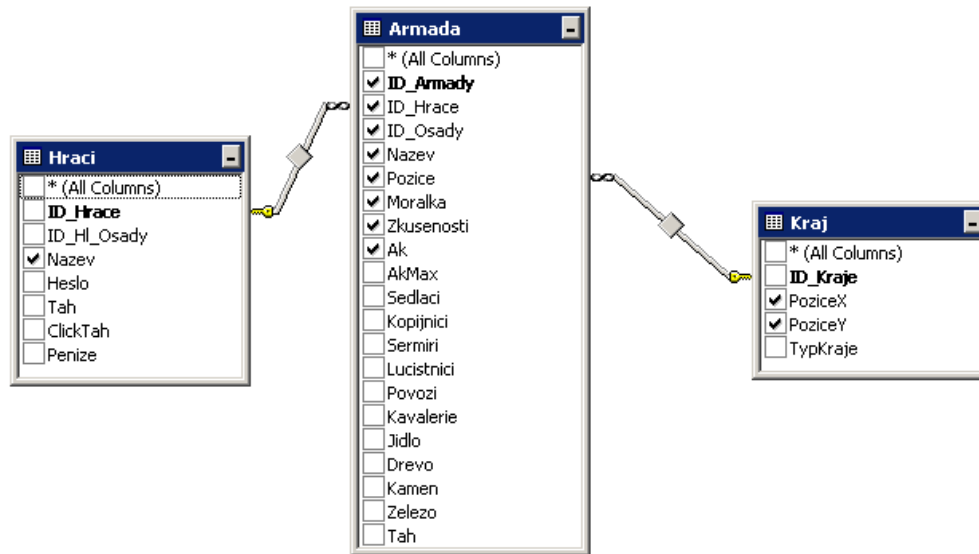
Pohled načítá z tabulek „Osada“, „Drevorubec“, „DulNaZezezo“, „Farma“, „Hrad“, „Kamenolom“, „Kasarna“, „Kostel“, „Opevneni“, „Radnice “ a „Sklad“ data, potřebná k sestavení objektu osada.

3.3.7. ViewOsadaFinance



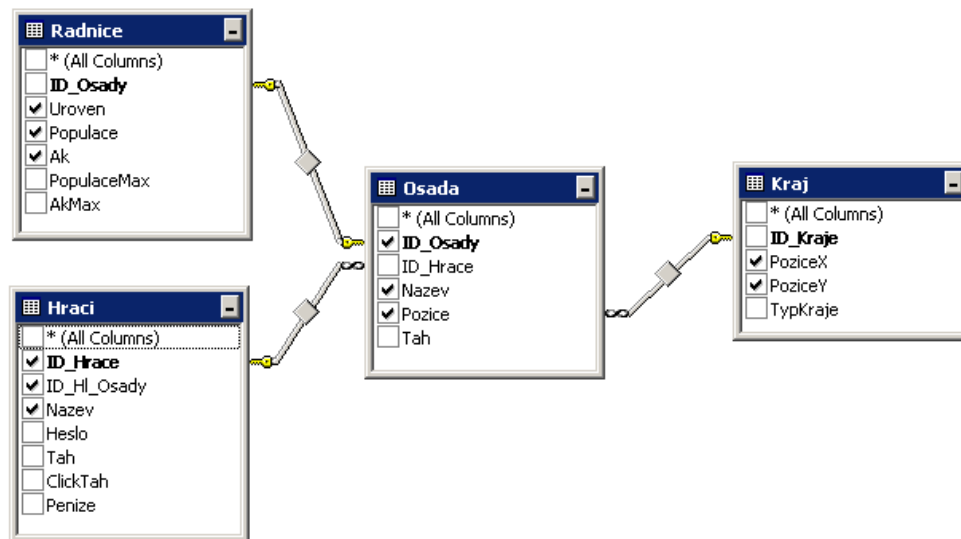
Pohled načítá z tabulek „Osada“, „Drevorubec“, „DulNaZezezo“, „Farma“, „Kamenolom“, „Kasarna“, „Opevneni“ a „Radnice “ data, potřebná k výpočtu příjmu osady.

3.3.8. ViewSeznamArmad



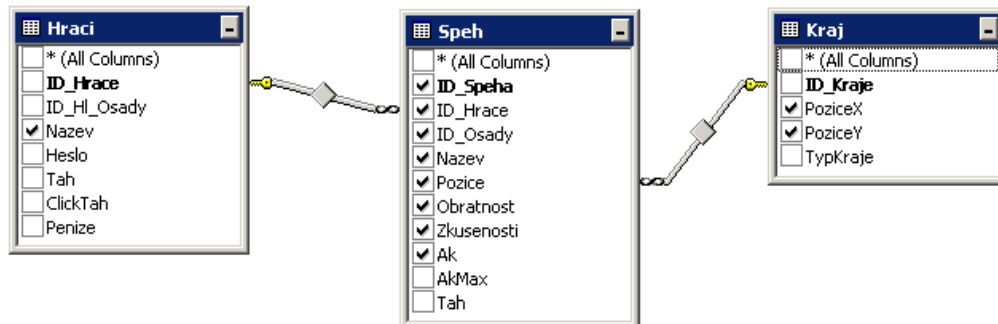
Pohled načítá z tabulek „Hraci“, „Armada“ a „Kraj“ data, potřebná k vytvoření seznamu armád patřících jednomu hráči.

3.3.9. ViewSeznamOsad



Pohled načítá z tabulek „Hraci“, „Osada“, „Radnice“ a „Kraj“ data, potřebná k vytvoření seznamu osad patřících jednomu hráči.

3.3.10. ViewSeznamSpehu



Pohled načítá z tabulek „Hraci“, „Speh“ a „Kraj“ data, potřebná k vytvoření seznamu špehů patřících jednomu hráči.

3.3.11. ViewZpravyDorucene



Pohled načítá z tabulek „Hraci“, „Dorucene“ a „Zpravy“ data, potřebná k vytvoření seznamu doručených zpráv vybranému hráči.

3.3.12. ViewZpravyOdeslane



Pohled načítá z tabulek „Hraci“, „Odeslane“ a „Zpravy“ data, potřebná k vytvoření seznamu odeslaných zpráv vybranému hráči.

3.4. Procedury

3.4.1. ArmadaNakupAk

- Vstupní parametry:
 - ID_Armady: integer
 - Pocet: integer
- Popis: Procedura zvýší v tabulce „Armada“ hodnotu atributu „AkMax“ o hodnotu parametru „Pocet“ a odečte za to o příslušný počet hodnotu atributu „Zkusenosti“. Změna se týká jen řádku určeného parametrem „ID_Armady“.

3.4.2. ArmadaNakupMoralky

- Vstupní parametry:
 - ID_Armady: integer
 - Pocet: integer
- Popis: Procedura zvýší v tabulce „Armada“ hodnotu atributu „Moralka“ o hodnotu parametru „Pocet“ a odečte zato o příslušný počet hodnotu atributu „Zkusenosti“. Změna se týká jen řádku určeného parametrem „ID_Armady“.

3.4.3. ArmadaNastavAk

- Vstupní parametry:
 - ID_Armady: integer
- Popis: Procedura nastaví v tabulce „Armada“ hodnotu atributu „Ak“ na hodnotu atributu „AkMax“. Změna se týká jen řádku určeného parametrem „ID_Armady“.

3.4.4. ArmadaOsekejSuroviny

- Vstupní parametry:
 - ID_ArmadyA: integer
 - Zatez: integer
- Popis: Procedura zkontroluje zda v tabulce „Armada“ nejsou hodnoty atributů „Jidlo“, „Drevo“, „Kamen“ a „Zezezo“ větší než je hodnota parametru „Zatez“. Pokud ano, změni je právě na tuto hodnotu. Změna se týká jen řádku určeného parametrem „ID_ArmadyA“.

3.4.5. ArmadaPresun

- Vstupní parametry:
 - ID_Armady: integer
 - Rychlost: integer
 - Pocet: integer
 - Kam: integer
- Popis: Procedura na základě parametru „Rychlost“ a „Pocet“ vypočte, kolik bude k přesunu armady na novou pozici potřeba akčních bodů. Pokud armáda bude mít dostatek akčních bodů, provede se přesun armády, tzn. v tabulce „Armada“ se změni hodnota atributu „Pozice“ na hodnotu parametru „Kam“. Změna se týká jen řádku určeného parametrem „ID_Armady“.

3.4.6. ArmadaPresunSurovin

- Vstupní parametry:
 - ID_ArmadyA: integer
 - ID_ArmadyB: integer
 - Jidlo: integer
 - Drevo: integer
 - Kamen: integer
 - Zezezo: integer
- Popis: Procedura přesune suroviny z armády A do armády B. Hodnoty parametrů „Jidlo“, „Drevo“, „Kamen“ a „Zezezo“ určují kolik příslušných surovin se má přesunout. Pokud tyto hodnoty budou záporné, přesouvají se vlastně suroviny z armády B do armády A. Parametrem „ID_ArmadyA“ určuje armádu A, parametrem „ID_ArmadyB“ určuje armádu B.

3.4.7. ArmadaPresunVojaku

- Vstupní parametry:
 - ID_ArmadyA: integer
 - ID_ArmadyB: integer
 - Sedlak: integer
 - Kopijnik: integer
 - Sermir: integer
 - Lucistnik: integer
 - Povez: integer
 - Kavalerie: integer
- Popis: Procedura přesune vojáky z armády A do armády B. Hodnoty parametrů „Sedlak“, „Kopijnik“, „Sermir“, „Lucistnik“, „Povez“ a „Kavalerie“ určují kolik vojáků se má přesunout. Pokud tyto hodnoty budou záporné, přesouvají se vlastně vojáci z armády B do armády A. Parametrem „ID_ArmadyA“ určuje armádu A, parametrem „ID_ArmadyB“ určuje armádu B.

3.4.8. HracTah

- Vstupní parametry:
 - ID_Hrace: integer
 - UdrzbaVsechVojaku: integer
- Popis: Procedura spouští proceduru „OsadaTah“ pro všechny osady daného hráče. Poté se pro daného hráče ve všech armádách a u všech špehů obnoví akční body na maximum. A nakonec se hráči odečte z pokladnice žold všech vojáků, které má v osadách a armádách. Pokud po odečtení peněz bude pokladnice v mínusu, přičte se úrok 10%. A pokud se peníze dostanou pod úroveň -10 000, hráč přijde o všechny vojáky ve všech osadách i armádách.

3.4.9. MensiZeDvou

- Vstupní parametry:
 - a: integer
 - b: integer
- Výstupní parametry:
 - a: integer
- Popis: Procedura ze dvou čísel vrací to menší.

3.4.10. OsadaDrevorubecNastavDelniky

- Vstupní parametry:
 - ID_Osady: integer
 - PocetDelniku: integer
- Popis: Procedura nastaví počet dělníků, kolik má v produkci pracovat, tento počet však nikdy nebude menší než je počet volných dělníků a maximum dělníků v produkci.

3.4.11. OsadaDrevorubecZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - DelniciMax: integer
 - Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“). Parametr „DelniciMax“ pak představuje maximum dělníků v produkci po zvýšení úrovně.

3.4.12. OsadaDulNaZezezoNastavDelniky

- Vstupní parametry:
 - ID_Osady: integer
 - PocetDelniku: integer
- Popis: Procedura nastaví počet dělníků, kolik má v produkci pracovat, tento počet však nikdy nebude menší než je počet volných dělníků a maximum dělníků v produkci.

3.4.13. OsadaDulNaZeLezoZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - DelniciMax: integer
 - Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“). Parametr „DelniciMax“ pak představuje maximum dělníků v produkci po zvýšení úrovně.

3.4.14. OsadaFarmaNastavDelniky

- Vstupní parametry:
 - ID_Osady: integer
 - PocetDelniku: integer
- Popis: Procedura nastaví počet dělníků, kolik má v produkci pracovat, tento počet však nikdy nebude menší než je počet volných dělníků a maximum dělníků v produkci.

3.4.15. OsadaFarmaZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - DelniciMax: integer
 - Krok: integer

- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“). Parametr „DelniciMax“ pak představuje maximum dělníků v produkci po zvýšení úrovně.

3.4.16. OsadaHradZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“).

3.4.17. OsadaKamenolomNastavDelniky

- Vstupní parametry:
 - ID_Osady: integer
 - PocetDelniku: integer
- Popis: Procedura nastaví počet dělníků, kolik má v produkci pracovat, tento počet však nikdy nebude menší než je počet volných dělníků a maximum dělníků v produkci.

3.4.18. OsadaKamenolomZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer

- DelniciMax: integer
- Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“). Parametr „DelniciMax“ pak představuje maximum dělníků v produkci po zvýšení úrovně.

3.4.19. OsadaKasarnaVycvicKavalerie

- Vstupní parametry:
 - ID_Osady: integer
 - Populace: integer
 - Ak: integer
 - Drevo: integer
 - Zelezo: integer
 - Pocet: integer
- Popis: Procedura přičte k atributu „Kavalerie“ tabulky „Kasarna“ příslušné osady, hodnotu parametru „Pocet“, ale jen pokud je k tomu ve skladě dostatek dřeva a železa, (což je dáno parametry „Drevo“ a „Zelezo“) a radnice má dostatek akčních bodů a volné populace (to je určeno parametrem „Ak“ a „Populace“).

3.4.20. OsadaKasarnaVycvicKopijniky

- Vstupní parametry:
 - ID_Osady: integer
 - Populace: integer
 - Ak: integer
 - Drevo: integer
 - Zelezo: integer
 - Pocet: integer
- Popis: Procedura přičte k atributu „Kopijnici“ tabulky „Kasarna“ příslušné osady, hodnotu parametru „Pocet“, ale jen pokud je k tomu ve skladě dostatek dřeva a železa, (což je dáno parametry „Drevo“ a „Zelezo“) a radnice má dostatek akčních bodů a volné populace (to je určeno parametrem „Ak“ a „Populace“).

3.4.21. OsadaKasarnaVycvicLucistniki

- Vstupní parametry:
 - ID_Osady: integer
 - Populace: integer
 - Ak: integer
 - Drevo: integer
 - Zelezo: integer
 - Pocet: integer
- Popis: Procedura přičte k atributu „Lucistniki“ tabulky „Kasarna“ příslušné osady, hodnotu parametru „Pocet“, ale jen pokud je k tomu ve skladě dostatek dřeva a železa, (což je dáno parametry „Drevo“ a „Zelezo“) a radnice má dostatek akčních bodů a volné populace (to je určeno parametrem „Ak“ a „Populace“).

3.4.22. OsadaKasarnaVycvicPovozi

- Vstupní parametry:
 - ID_Osady: integer
 - Populace: integer
 - Ak: integer
 - Drevo: integer
 - Zelezo: integer
 - Pocet: integer
- Popis: Procedura přičte k atributu „Povozi“ tabulky „Kasarna“ příslušné osady, hodnotu parametru „Pocet“, ale jen pokud je k tomu ve skladě dostatek dřeva a železa, (což je dáno parametry „Drevo“ a „Zelezo“) a radnice má dostatek akčních bodů a volné populace (to je určeno parametrem „Ak“ a „Populace“).

3.4.23. OsadaKasarnaVycvicSedlaky

- Vstupní parametry:
 - ID_Osady: integer
 - Populace: integer
 - Ak: integer

- Drevo: integer
- Zelezo: integer
- Pocet: integer
- Popis: Procedura přičte k atributu „Sedlaci“ tabulky „Kasarna“ příslušné osady, hodnotu parametru „Pocet“, ale jen pokud je k tomu ve skladě dostatek dřeva a železa, (což je dáno parametry „Drevo“ a „Zelezo“) a radnice má dostatek akčních bodů a volné populace (to je určeno parametrem „Ak“ a „Populace“).

3.4.24. OsadaKasarnaVycvicSermire

- Vstupní parametry:
 - ID_Osady: integer
 - Populace: integer
 - Ak: integer
 - Drevo: integer
 - Zelezo: integer
 - Pocet: integer
- Popis: Procedura přičte k atributu „Sermiri“ tabulky „Kasarna“ příslušné osady, hodnotu parametru „Pocet“, ale jen pokud je k tomu ve skladě dostatek dřeva a železa, (což je dáno parametry „Drevo“ a „Zelezo“) a radnice má dostatek akčních bodů a volné populace (to je určeno parametrem „Ak“ a „Populace“).

3.4.25. OsadaKasarnaZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“).

3.4.26. OsadaKostelZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - Spokojenost: integer
 - Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“). Parametr „Spokojenost“ pak představuje spokojenost v osadě po zvýšení úrovně kostela.

3.4.27. OsadaNastavAk

- Vstupní parametry:
 - ID_Osady: integer
- Popis: Ve vybrané osadě nastaví akční body na maximum.

3.4.28. OsadaOpevneniZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“).

3.4.29. OsadaPopulace

- Vstupní parametry:
 - ID_Osady: integer
- Výstupní parametry:
 - PočetPopulace: integer
- Popis: Procedura vrací aktuální stav populace ve vybrané osadě.

3.4.30. OsadaPopulaceVolna

- Vstupní parametry:
 - ID_Osady: integer
- Výstupní parametry:
 - PočetVolnéPopulace: integer
- Popis: Procedura vrací aktuální stav volné populace ve vybrané osadě, tj. populace, která není zaměstnaná v žádné z produkcí.

3.4.31. OsadaPopulaceVProdukcí

- Vstupní parametry:
 - ID_Osady: integer
- Výstupní parametry:
 - PočetVolnéPopulace: integer
- Popis: Procedura vrací aktuální stav populace zaměstnané v kterékoliv z produkcí vybrané osady.

3.4.32. OsadaPresunSurovin

- Vstupní parametry:
 - ID_Armady: integer
 - ID_Osady: integer
 - Jidlo: integer
 - Drevo: integer

- Kamen: integer
- Zelezo: integer
- Popis: Procedura přesune suroviny z armády do osady. Hodnoty parametrů „Jidlo“, „Drevo“, „Kamen“ a „Zelezo“ určují kolik příslušných surovin se má přesunout. Pokud tyto hodnoty budou záporné, přesouvají se vlastně suroviny z osady do armády.

3.4.33. OsadaPresunVojaku

- Vstupní parametry:
 - ID_Armady: integer
 - ID_Osady: integer
 - Sedlak: integer
 - Kopijnik: integer
 - Sermir: integer
 - Lucistnik: integer
 - Povez: integer
 - Kavalerie: integer
- Popis: Procedura přesune vojáky z armády do osady. Hodnoty parametrů „Sedlak“, „Kopijnik“, „Sermir“, „Lucistnik“, „Povez“ a „Kavalerie“ určují kolik vojáků se má přesunout. Pokud tyto hodnoty budou záporné, přesouvají se vlastně vojáci z osady do armády.

3.4.34. OsadaPrirustekSurovin

- Vstupní parametry:
 - ID_Osady: integer
- Popis: Procedura přičte ve skladě suroviny z produkce, tato procedura je volaná z procedury „OsadaTah“.

3.4.35. OsadaPrirustekZlata

- Vstupní parametry:
 - ID_Osady: integer
- Popis: Procedura přičte hráči peníze podle volné populace vybrané osady, tato procedura je volaná z procedury „OsadaTah“.

3.4.36. OsadaRadniceZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - PopulaceMax: integer
 - AkMax: integer
 - Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“). Parametry „PopulaceMax“ a „AkMax“ pak představuje maximální možnou populaci a maximum akčních bodů v osadě po zvýšení úrovně Radnice.

3.4.37. OsadaSkladZmenaUrovne

- Vstupní parametry:
 - ID_Osady: integer
 - Ak: integer
 - Drevo: integer
 - Kamen: integer
 - SurovinMax: integer
 - Krok: integer
- Popis: Procedura zvýší (sníží) úroveň budovy o jedna, pokud je hodnota parametru „Krok“ kladná (záporná). Zvýšení nastane jen pokud je k tomu ve skladě dostatek dřeva a kamene, (což je dáno parametry „Drevo“ a „Kamen“) a radnice má dostatek akčních bodů (to určuje parametr „Ak“). Parametr „SurovinMax“ pak představuje maximum surovin v osadě po zvýšení úrovně skladu.

3.4.38. OsadaTah

- Vstupní parametry:
 - ID_Osady: integer
- Popis: Procedura provede ve vybrané osadě všechny potřebné kroky, aby mohl nastat nový tah. Z této procedury jsou volané následující procedury „OsadaNastavAK“, „OsadaPrirustekZlata“, „OsadaPrirustekSurovin“ a „OsadaZvysitPopulaci“.

3.4.39. OsadaZvysitPopulace

- Vstupní parametry:
 - ID_Osady: integer
- Popis: Procedura ve vybrané osadě přičte nebo odečte populaci a to v závislosti na spokojenosti v osadě. Tato procedura je volaná z procedury „OsadaTah“.

3.4.40. SpehNakupAk

- Vstupní parametry:
 - ID_Speha: integer
 - Pocet: integer
- Popis: Procedura zvýší v tabulce „Speh“ hodnotu atributu „AkMax“ o hodnotu parametru „Pocet“ a odečte zato o příslušný počet hodnotu atributu „Zkusenosti“. Změna se týká jen řádku určeného parametrem „ID_Speha“.

3.4.41. SpehNakupObratnost

- Vstupní parametry:
 - ID_Speha: integer
 - Pocet: integer
- Popis: Procedura zvýší v tabulce „Speh“ hodnotu atributu „Obratnost“ o hodnotu parametru „Pocet“ a odečte zato o příslušný počet hodnotu atributu „Zkusenosti“. Změna se týká jen řádku určeného parametrem „ID_Speha“.

3.4.42. SpehNastavAk

- Vstupní parametry:
 - ID_Speha: integer
- Popis: Procedura nastaví v tabulce „Speh“ hodnotu atributu „Ak“ na hodnotu atributu „AkMax“. Změna se týká jen řádku určeného parametrem „ID_Speha“.

3.4.43. SpehPresun

- Vstupní parametry:
 - ID_Speha: integer
 - Rychlost: integer
 - Pocet: integer
 - Kam: integer
- Popis: Procedura na základě parametru „Rychlost“ a „Pocet“ vypočte, kolik bude k přesunu špeha na novou pozici potřeba akčních bodů. Pokud špeh bude mít dostatek akčních bodů, provede se přesun špeha tzn. v tabulce „Speh“ se změní hodnota atributu „Pozice“ na hodnotu parametru „Kam“. Změna se týká jen řádku určeného parametrem „ID_Speha“.

3.4.44. SvetClickNaTahZbyva

- Výstupní parametry:
 - KolikAktivníchHracuNedaloKlikTah: integer
- Popis: Procedura vrací kolik aktivních hráčů nedalo povolení k předčasnému dokončení aktuálního tahu hry.

3.4.45. SvetPocetTahu

- Výstupní parametry:
 - PocetTahu: integer
- Popis: Procedura podle aktuálního času a datumu vrací, kolik se má provést tahů hry. Pokud se žádný z hráčů delší dobu nepřipojí, neprovádí se žádné tahy, v momentě kdy se jeden hráč připojí, tak tato procedura vrátí kolik tahů se má provést.

3.4.46. SvetTah

- Výstupní parametry:
 - PocetTahu: integer
- Popis: Procedura podle aktuálního času a datumu nebo podle povolení k předčasnému dokončení aktuálního tahu všemi aktivními hráči, provede zvýšení tahu hry.

3.4.47. VytvorKraje

- Vstupní parametry:
 - x: integer
 - y: integer
- Popis: Do tabulky „Kraj“ vloží $x * y$ záznamů s hodnotou atributu „Typ-Kraje“ rovnu „1“ (terén tráva), čímž vytvoří hrací plochu o velikosti $x * y$.

3.4.48. ZpravaPosli

- Vstupní parametry:
 - ID_Kdo: integer
 - Komu: string
 - Nadpis: string
 - Text: string
- Popis: Procedura pošle zprávu, příjemce je zadán jménem.

3.4.49. ZpravaPosliId

- Vstupní parametry:
 - ID_Kdo: integer
 - Komu: integer
 - Nadpis: string
 - Text: string
- Popis: Procedura pošle zprávu, příjemce je zadán identifikačním číslem.

4. Seznam tříd

4.1. Armada

Dědí z třídy „Jednotka“.

- Atributy: Třída má kromě zděděných atributů ještě tyto.

Atribut	Datový typ	Popis
jidloSkladem	int	Hodnota udává kolik jídla armáda nese.
drevoSkladem	int	Hodnota udává kolik dřeva armáda nese.
kamenSkladem	int	Hodnota udává kolik kamene armáda nese.
zelezoSkladem	int	Hodnota udává kolik železa armáda nese.
sedlak	int	Hodnota udává kolik sedláků je v armádě.
kopijnik	int	Hodnota udává kolik kopijníků je v armádě.
sermir	int	Hodnota udává kolik šermířů je v armádě.
lucistnik	int	Hodnota udává kolik lučištníků je v armádě.
povoz	int	Hodnota udává kolik povozů je v armádě.
kavalerie	int	Hodnota udává kolik jezdců je v armádě.
moralka	int	Hodnota udává sílu morálky v armádě.

- Vlastnosti: Třída má kromě zděděných Vlastností ještě tyto.

Vlastnost	Datový typ	Čtení/zápis	Popis
Jidlo	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „jidloSkladem“.
Drevo	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „drevoSkladem“.
Kamen	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „kamenSkladem“.
Zezezo	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „zezezoSkladem“.
Sedlaci	int	Ano/Ne	Vrací hodnotu atributu „sedlak“.
Kopijnici	int	Ano/Ne	Vrací hodnotu atributu „kopijnik“.
Sermiri	int	Ano/Ne	Vrací hodnotu atributu „sermir“.
Lucistnici	int	Ano/Ne	Vrací hodnotu atributu „lucistnik“.
Povozi	int	Ano/Ne	Vrací hodnotu atributu „povoz“.
Kavalerie	int	Ano/Ne	Vrací hodnotu atributu „kavalerie“.
Moralka	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „moralka“.
Obrana	int	Ano/Ne	Vrací obrannou sílu celé armády.
Utok	int	Ano/Ne	Vrací útočnou sílu celé armády.
Rychlost	int	Ano/Ne	Vrací kolik akčních bodů potřebuje armáda k přesunu o jedno políčko na mapě.
Zatez	int	Ano/Ne	Vrací kolik surovin celá armáda unese.
Zold	int	Ano/Ne	Vrací součet žoldů všech vojáků v armádě.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

Metoda	Vrací	Popis
Armada	nic	Jedná se o konstruktor třídy a je jednou přetížen.

4.2. ArmadaInfo

Shromažďuje část informací o armádě. Využití např. k zobrazení seznamu armád, kdy se nemusí načítat z databáze všechny údaje o každé armádě, ale načtou se jen vybraná data.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
Ak	int	Hodnota udává aktuální počet akčních bodů armády.
ID_Armady	int	Hodnota udává identifikační číslo armády.
ID_Hrace	int	Hodnota udává identifikační číslo hráče, kterému armáda patří.
ID_Osady	int	Hodnota udává identifikační číslo osady v které se armáda nachází, pokud je rovna nule, znamená to, že je mimo osadu.
JmenoHrace	string	Hodnota udává jméno hráče, kterému armáda patří.
Moralka	int	Hodnota udává morálku v armádě.
NazevArmady	string	Hodnota udává název armády.
Pozice	int	Hodnota udává identifikační číslo kraje (políčka na mapě) na kterém se armáda nachází.
PoziceX	int	Hodnota udává ixovou souřadnici pozice.
PoziceY	int	Hodnota udává ypsilonovou souřadnici pozice.
Zkusenosti	int	Hodnota udává počet zkušenostních bodů armády.

- Vlastnosti: Třída má následující Vlastnosti.

Vlastnost	Datový typ	Čtení/zápis	Popis
ak	int	Ano/Ne	Vrací hodnotu atributu „Ak“.
Id_Armady	int	Ano/Ne	Vrací hodnotu atributu „ID_Armady“.
Id_Hrace	int	Ano/Ne	Vrací hodnotu atributu „ID_Hrace“.
Id_Osady	int	Ano/Ne	Vrací hodnotu atributu „ID_Osady“.
jmenoHrace	int	Ano/Ne	Vrací hodnotu atributu „JmenoHrace“.
moralka	int	Ano/Ne	Vrací hodnotu atributu „Moralka“.
nazevArmady	int	Ano/Ne	Vrací hodnotu atributu „NazevArmady“.
pozice	int	Ano/Ne	Vrací hodnotu atributu „Pozice“.
poziceX	int	Ano/Ne	Vrací hodnotu atributu „PoziceX“.
poziceY	int	Ano/Ne	Vrací hodnotu atributu „PoziceY“.
zkusenosti	int	Ano/Ne	Vrací hodnotu atributu „Zkusenosti“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

Metoda	Vrací	Popis
ArmadaInfo	nic	Jedná se o konstruktor třídy a je jednou přetížen.

4.3. Armady

Jedná se o statickou třídu. Slouží k načítání a editaci objektů třídy „Armada“ a „Obrance“. Třída má následující statické metody.

- BourejVsechnyBudovy
 - Vstupní parametry:
 - * o: Objekt třídy „Obrance“
 - * a: Objekt třídy „Armada“
 - Vrací: SQL příkaz
 - Popis: Soukromá statická metoda volaná v případě prohry obráncem z metody „PrepocitejVlastnostiASuroviny“ a vrací řetězec SQL příkazů,

které v obráncově osadě sníží úroveň všech budov o jedna, maximálně však na úroveň jedna.

- DruhaVlna
 - Vstupní parametry:
 - * o: Objekt třídy „Obrance“
 - * u: Objekt třídy „Armada“
 - Vrací: Celé číslo
 - Popis: Soukromá statická metoda, volaná z metody „PrvniVlna“, vrací celé číslo představující výsledek bitvy.
- Odkryj
 - Vstupní parametry:
 - * id_armady: int
 - Popis: Veřejná statická metoda, která v databázi nastaví hodnotu atributu „ID_Osady“ v tabulce „Armada“ na „0“ a tím objekt armáda v podstatě opustí úkryt osady, čímž se armáda bude zobrazovat na mapě.
- MensiZeDvou
 - Vstupní parametry:
 - * a: int
 - * b: int
 - Vrací: int
 - Popis: Soukromá statická metoda volaná z více metod, přebírá dvě celá čísla a vrací to větší.
- MuzeSeSkryt
 - Vstupní parametry:
 - * u: Objekt třídy „Armada“
 - Vrací: int
 - Popis: Veřejná statická metoda, která vrací id osady, do které se může armáda ukrýt, pokud v danou chvíli taková osada neexistuje vrací „0“.
- NactiArmadu
 - Vstupní parametry:
 - * id_armady: int

- Vrací: Objekt třídy „Armada“
- Popis: Veřejná statická metoda, která na základě id armády načte z databáze data a z nich vytvoří objekt třídy „Armada“.
- NactiArmady
 - Vstupní parametry:
 - * id_hrace: int
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda, která na základě id hráče načte z databáze data a z nich vytvoří objekty třídy uvArmadaInfo a vloží je do pole.
- NactiInfoOArmade
 - Vstupní parametry:
 - * id_armady: int
 - Vrací: Objekt třídy „ArmadaInfo“
 - Popis: Veřejná statická metoda, která na základě id armády načte z databáze data a z nich vytvoří objekt třídy uvArmadaInfo.
- NactiObrance
 - Vstupní parametry:
 - * id_armady: int
 - Vrací: Objekt třídy „Obrance“
 - Popis: Veřejná statická metoda, která na základě id armády načte z databáze data a z nich vytvoří objekt třídy uvObrance.
- ObjektyKUtoku
 - Vstupní parametry:
 - * a: Objekt třídy „Armada“
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda, která pro objekt třídy „Armada“ načte z databáze id a název všech objektů na které může být veden útok.
- ObjektyKVymeneSurovin
 - Vstupní parametry:
 - * a: Objekt třídy „Armada“

- Vrací: ArrayList
- Popis: Veřejná statická metoda, která pro objekt třídy „Armada“ načte z databáze id a název všech objektů s kterými může vyměňovat suroviny.
- ObjektyKVymeneVojaku
 - Vstupní parametry:
 - * a: Objekt třídy „Armada“
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda, která pro objekt třídy uvArmada načte z databáze id a název všech objektů s kterými může vyměňovat vojáky.
- Prejmenuj
 - Vstupní parametry:
 - * a: Objekt třídy „Armada“
 - * jmeno: string
 - Vrací: void
 - Popis: Veřejná statická metoda, která objektu třídy „Armada“ změní hodnotu atributu „Nazev“.
- PrepocetSurovin
 - Vstupní parametry:
 - * u: Objekt třídy „Armada“
 - * o: Objekt třídy „Obrance“
 - Vrací: string
 - Popis: Soukromá statická metoda volaná v případě prohry obránce z metody „PrepVlastASur“ a vrací řetězec SQL příkazů, které přesunou suroviny od obránce k útočníkovi.
- PrepocitejVlastnostiASuroviny
 - Vstupní parametry:
 - * vysledek: int
 - * u: Objekt třídy „Armada“
 - * o: Objekt třídy „Obrance“
 - Vrací: string

- Popis: Soukromá statická metoda volaná z metody „Utok“ a vrací řetězec SQL příkazů, které upraví v databázi informace podle výsledku bitvy.
- Presun
 - Vstupní parametry:
 - * id_Hrace: int
 - * id_Armady: int
 - * kam: int
 - Vrací: void
 - Popis: Veřejná statická metoda, která přesune objekt třídy „Armada“ na pozici na mapě určenou parametrem „kam“.
- PrvniVlna
 - Vstupní parametry:
 - * u: Objekt třídy „Armada“
 - * o: Objekt třídy „Obrance“
 - Vrací: int
 - Popis: Soukromá statická metoda volaná z metody „Utok“ a vrací celé číslo reprezentující výsledek bitvy, metoda provede výpočet bitvy a rozhodne jestli nastane druhé kolo bitvy nebo bitva končí po prvním kole.
- Skryj
 - Vstupní parametry:
 - * id_armady: int
 - Popis: Veřejná statická metoda, která v databázi nastaví hodnotu atributu „ID_Osady“ v tabulce „Armada“ na id osady, kde chceme armádu skrýt. Skrýt lze armádu jen pokud je na stejném políčku jako osada, kde se má ukrýt. Tato osada musí samozřejmě patřit vlastníkově armády a armáda musí být bez vojáků.
- Utok
 - Vstupní parametry:
 - * u: Objekt třídy „Armada“
 - * o: Objekt třídy „Obrance“
 - Vrací: string

- Popis: Veřejná statická metoda, provede výpočet bitvy mezi útočníkem a obráncem, i s úpravou databáze. Vrací řetězec s výsledkem bitvy.
- UtokDruheVlny
 - Vstupní parametry:
 - * druhuVlnaObrana: double
 - * a: Objekt třídy „Obrance“
 - Vrací: void
 - Popis: Soukromá statická metoda volaná z metody „DruhaVlna“, parametr „druhaVlnaObrana“ určuje kolik zbylo obránci obranné síly a na základě toho provede úpravu objektu třídy „Obrance“.
- UtokDruheVlny (přetížení)
 - Vstupní parametry:
 - * druhuVlnaObrana: double
 - * a: Objekt třídy „Armada“
 - Vrací: void
 - Popis: Soukromá statická metoda volaná z metody „DruhaVlna“, parametr „druhaVlnaObrana“ určuje kolik zbylo útočníkovi obranné síly a na základě toho provede úpravu objektu třídy „Armada“.
- UtokPrvniVlny
 - Vstupní parametry:
 - * prvniVlnaObrana: double
 - * a: Objekt třídy „Obrance“
 - Vrací: void
 - Popis: Soukromá statická metoda volaná z metody „PrvniVlna“, parametr „prvniVlnaObrana“ určuje kolik zbylo obránci obranné síly a na základě toho provede úpravu objektu třídy „Obrance“.
- UtokPrvniVlny (přetížení)
 - Vstupní parametry:
 - * prvniVlnaObrana: double
 - * a: Objekt třídy „Armada“
 - Vrací: void

- Popis: Soukromá statická metoda volaná z metody „PrvniVlna“, parametr „prvniVlnaObrana“ určuje kolik zbylo utočnickovy obranné síly a na základě toho provede úpravu objektu třídy „Armada“.
- VysledekBitvy
 - Vstupní parametry:
 - * vysledek: int
 - * u: Objekt třídy „Armada“
 - * o : Objekt třídy „Obrance“
 - Vrací: string
 - Popis: Soukromá statická metoda volaná z metody „Utok“. Na základě parametru „vysledek“ metoda pozná, jak bitva dopadla a s pomocí objektů útočníka a obrance vytvoří řetězec s textem o výsledku bitvy.
- VytvorArmadu
 - Vstupní parametry:
 - * o: Objekt třídy „Osada“
 - * jmeno: string
 - Vrací: void
 - Popis: Veřejná statická metoda. V osadě vytvoří objekt třídy „Armada“ s názvem podle parametrů „jmeno“.
- ZabitychObrancu
 - Vstupní parametry:
 - * o : Objekt třídy „Obrance“
 - Vrací: int
 - Popis: Soukromá statická metoda volaná z metody „PrepocitejVlastnostiASuroviny“. Vrací kolik bylo při bitvě zabitých obránců.
- ZabitychUtocniku
 - Vstupní parametry:
 - * u: Objekt třídy „Armada“
 - Vrací: int
 - Popis: Soukromá statická metoda volaná z metody „PrepocitejVlastnostiASuroviny“. Vrací kolik bylo při bitvě zabitých útočníků.

4.4. Budova

Od této třídy dědí všechny budovy v objektu třídy Osada.

- Atributy: Třída má následující čtyři atributy.

Atribut	Datový typ	Popis
ak	int	Hodnota udává kolik je potřeba akčních bodů ke zvýšení úrovně budovy o jedna.
drevo	int	Hodnota udává kolik je potřeba dřeva ke zvýšení úrovně budovy o jedna.
kamen	int	Hodnota udává kolik je potřeba kamene ke zvýšení úrovně budovy o jedna.
uroven	int	Hodnota určuje úroveň budovy.

- Vlastnosti: Třída má následující čtyři Vlastnosti.

Vlastnost	Datový typ	Čtení/zápis	Popis
Ak	int	Ano/Ne	Vrací hodnotu atributu „ak“.
Drevo	int	Ano/Ne	Vrací hodnotu atributu „drevo“.
Kamen	int	Ano/Ne	Vrací hodnotu atributu „kamen“.
Uroven	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „uroven“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Budova

* Vstupní parametry:

· Uroven: int

* Vrací: void

* Popis: Jedná se o konstruktor třídy a je jednou přetížen.

4.5. databáze

Jedná se o statickou třídu. Slouží ke komunikaci s databází. Obsahuje metody pro vytvoření a opravu databázového systému. Dále se zde nacházejí metody pro vytvoření, ověření a editaci hráčských účtů.

- Atributy: Třída má jediný veřejný statický atribut.

Atribut	Datový typ	Popis
conn	SqlConnection	String sloužící ke spojení s databází.

- Metody: Třída má následující statické metody.
 - JmenoHrace
 - * Vstupní parametry:
 - id_hrace: int
 - * Vrací: string
 - * Popis: Veřejná statická metoda, která podle id hráče vrací jeho jméno.
 - OverHrace
 - * Vstupní parametry:
 - jmeno: string
 - heslo: string
 - * Vrací: int
 - * Popis: Veřejná statická metoda, která vrací id hráče podle jména a hesla, pokud takový hráč neexistuje vrací „0“.
 - ProvedEditaci
 - * Vstupní parametry:
 - prikaz: string
 - * Vrací: void
 - * Popis: Veřejná statická metoda. Slouží k provádění SQL příkazů v databázi, přebírá tedy SQL příkaz a odesílá jej do databáze.
 - VymazPohledy
 - * Vrací: void
 - * Popis: Veřejná statická metoda, slouží k vymazání všech pohledů v databázovém systému hry.
 - VymazProcedury
 - * Vrací: void
 - * Popis: Veřejná statická metoda, slouží k vymazání všech procedur v databázovém systému hry.
 - VymazTriggery
 - * Vrací: void
 - * Popis: Veřejná statická metoda, slouží k vymazání všech triggerů v databázovém systému hry.
 - VytvorAdmin
 - * Vrací: void
 - * Popis: Veřejná statická metoda. Vytvoří v databázi tabulku „Admin“ a vloží do ní defaultního administrátora.

- VytvorPohledy
 - * Vrací: void
 - * Popis: Veřejná statická metoda, slouží k vytvoření všech pohledů v databázovém systému hry.
- VytvorProcedury
 - * Vrací: void
 - * Popis: Veřejná statická metoda, slouží k vytvoření všech procedur v databázovém systému hry.
- VytvorTabulky
 - * Vrací: void
 - * Popis: Veřejná statická metoda, slouží k vytvoření všech tabulek (kromě tabulky „Admin“) v databázovém systému hry. Vytvořené tabulky budou prázdné.
- VytvorTriggery
 - * Vrací: void
 - * Popis: Veřejná statická metoda, slouží k vytvoření všech triggerů v databázovém systému hry.
- ZaregistrujHrace
 - * Vstupní parametry:
 - název: string
 - pass: string
 - * Vrací: int
 - * Popis: Veřejná statická metoda, slouží k vytvoření nového hráče. Do tabulky „Hraci“ se přidá nový záznam kde parametr „nazev“ určuje jméno a parametr „pass“ heslo hráče. Nový záznam je vytvořen jen tehdy, pokud neexistuje již jiný hráč se stejným jménem a počet všech hráčů je menší než maximum počtu hráčů, nastavený ve hře.
- ZmenHrace
 - * Vstupní parametry:
 - id: int
 - nazev: string
 - pass: string
 - * Vrací: bool
 - * Popis: Veřejná statická metoda, slouží ke změně údajů o hráči. V tabulce „Hraci“ se upraví jméno podle parametrů „nazev“ a heslo podle parametr „pass“. Úprava se provede jen tehdy, pokud neexistuje již jiný hráč se stejným jménem.

4.6. Dorucenka

Třída slouží k manipulaci se zprávami.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
id_Zpravy	int	Hodnota udává identifikační číslo zprávy.
nadpis	string	Hodnota udává název zprávy.
datum	DateTime	Hodnota udává datum a čas vzniku zprávy.
odesilatel	int	Hodnota udává identifikační číslo odesílatele.
prijemce	int	Hodnota udává identifikační číslo příjemce.
jmeno	string	Hodnota udává jméno odesílatele pokud se jedná o doručenu zprávu nebo jméno příjemce jdeli o odeslanou zprávu.

- Vlastnosti: Třída má následující čtyři Vlastnosti.

Vlastnost	Datový typ	Čtení/zápis	Popis
Id_Zpravy	int	Ano/Ne	Vrací hodnotu atributu „id_Zpravy“.
nadpis	string	Ano/Ne	Vrací hodnotu atributu „nadpis“.
Datum	DateTime	Ano/Ne	Vrací hodnotu atributu „datum“.
Odesilatel	int	Ano/Ne	Vrací hodnotu atributu „odesilatel“.
Prijemce	int	Ano/Ne	Vrací hodnotu atributu „prijemce“.
Jmeno	string	Ano/Ne	Vrací hodnotu atributu „jmeno“.

Třída má jedinou metodu a tou je její konstruktor.

- Dorucenka
 - Vstupní parametry:
 - * id_Zpravy: int
 - * text: string
 - * kdy: DateTime
 - * odkoho: int
 - * komu: int

- * name: string
- Vrací: void
- Popis: Jedná se o konstruktor třídy a je jednou přetížen.

4.7. Drevorubec

Dědí z třídy „Produkce“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Drevorubec
 - * Vstupní parametry:
 - uroven: int
 - delnici: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.8. DulNaZezezo

Dědí z třídy „Produkce“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - DulNaZezezo
 - * Vstupní parametry:
 - uroven: int
 - delnici: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.9. Farma

Dědí z třídy „Produkce“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Farma
 - * Vstupní parametry:
 - uroven: int
 - delnici: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.10. Finance

Jedná se o statickou třídu. Slouží k získání přehledu příjmů a výdajů hráče. Třída má následující statické metody.

- ArmadyFinance
 - Vstupní parametry:
 - * id_Hrace: int
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Metoda ukládá do pole název a žold (v tomto pořadí) všech armád vybraného hráče. Tzn. hodnota v poli s indexem 0 je název a, s indexem 1 je žold armády atd..
- OsadyFinance
 - Vstupní parametry:
 - * id_Hrace: int
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Metoda ukládá do pole název, hrubý příjem, náklady na těžbu, žold domobrany a čistý příjem osady (v tomto pořadí) všech osad vybraného hráče. Tzn. hodnota v poli s indexem 0 je název, s indexem 1 je hrubý příjem, index 2 náklady na těžbu, index 3 žold domobrany a index 4 čistý příjem osady atd..
- StavPokladnice

- Vstupní parametry:
 - * id_Hrace: int
- Vrací: int
- Popis: Veřejná statická metoda, vrací aktuální stav peněz vybraného hráče.
- ZjistiNakladyNaVojaky
 - Vstupní parametry:
 - * id_Hrace: int
 - Vrací: int
 - Popis: Veřejná statická metoda, vrací žold všech vojáků ve všech osadách i armádách vybraného hráče.

4.11. Hrad

Dědí z třídy „Budova“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Hrad
 - * Vstupní parametry:
 - uroven: int
 - delnici: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.12. Jednotka

Třída slouží k dědění pro třídy „Speh“ a „Armada“.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
ID	int	Identifikační číslo armády nebo špeha.
ID_Hrace	int	Identifikační číslo hráče kterému jednotka patří.
ID_Osady	int	Identifikační číslo osady kde se jednotka nachází, pokud je mimo osadu je rovno „0“.
nazev	string	Jmeno armády nebo špeha.
pozice	int	Pozice jednotky na mapě.
zkusenosti	int	Aktuální počet zkušeností armády nebo špeha.
ak	int	Aktuální počet akčních bodů armády nebo špeha.
maxAk	int	Maximální počet akčních bodů armády nebo špeha.
tah	int	Číslo tahu hry kdy byla jednotka naposledy používána.

- Vlastnosti: Třída má následující vlastnosti.

Vlastnost	Datový typ	Čtení/zápis	Popis
iD	int	Ano/Ne	Vrací hodnotu atributu „iD“.
iD_Hrace	int	Ano/Ne	Vrací hodnotu atributu „iD_Hrace“.
iD_Osady	int	Ano/Ne	Vrací hodnotu atributu „iD_Osady“.
Nazev	int	Ano/Ne	Vrací hodnotu atributu „nazev“.
Pozice	int	Ano/Ne	Vrací hodnotu atributu „pozice“.
Zkusenosti	int	Ano/Ne	Vrací hodnotu atributu „zkusenosti“.
Ak	int	Ano/Ne	Vrací hodnotu atributu „ak“.
MaxAk	int	Ano/Ne	Vrací hodnotu atributu „maxAk“.
Tah	int	Ano/Ne	Vrací hodnotu atributu „tah“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Jednotka
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.13. Kamenolom

Dědí z třídy „Produkce“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Kamenolom

- * Vstupní parametry:
 - uroven: int
 - delnici: int
- * Vrací: void
- * Popis: Jedná se o konstruktor třídy.

4.14. Kasarna

Dědí z třídy „Budova“.

- Atributy: Třída má kromě zděděných atributů ještě tyto.

Atribut	Datový typ	Popis
sedlak	int	Aktuální počet sedláků v kasárnách.
kopijnik	int	Aktuální počet kopijníků v kasárnách.
sermir	int	Aktuální počet šermířů v kasárnách.
lucistnik	int	Aktuální počet lučištníků v kasárnách.
povoz	int	Aktuální počet povozů v kasárnách.
kavalerie	int	Aktuální počet kavalérie v kasárnách.
maxVojaci	int	Maximální počet vojáků v kasárnách.

- Vlastnosti: Třída má kromě zděděných Vlastností ještě tyto.

Vlastnost	Datový typ	Čtení/zápis	Popis
Sedlak	int	Ano/Ne	Vrací hodnotu atributu „sedlak“.
Kopijnik	int	Ano/Ne	Vrací hodnotu atributu „kopijnik“.
Sermir	int	Ano/Ne	Vrací hodnotu atributu „sermir“.
Lucistnik	int	Ano/Ne	Vrací hodnotu atributu „lucistnik“.
Povoz	int	Ano/Ne	Vrací hodnotu atributu „povoz“.
Kavalerie	int	Ano/Ne	Vrací hodnotu atributu „kavalerie“.
MaxVojaci	int	Ano/Ne	Vrací hodnotu atributu „maxVojaci“.
Obrana	int	Ano/Ne	Vrací celkovou obranou sílu všech vojáků v kasárnách.
Utok	int	Ano/Ne	Vrací celkovou utočnou sílu všech vojáků v kasárnách.

- Metody: Třída má následující metody.

- Kasárna
 - * Vstupní parametry:
 - uroven: int
 - Sedlak: int
 - Kopijnik: int
 - Sermir: int
 - Lucistnik: int
 - Povož: int
 - Kavalerie: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.
- Posadka
 - * Vrací: int
 - * Popis: Vrací kolik populace v kasárnách zabírají všichni vojáci.
- VlastnostiJednotek
 - * Vrací: ArrayList
 - * Popis: Vrací formátované pole se seznamem vlastností jednotek. Pole má velikost 6 * 12. 6 je druhů jednotek a od každé jednotky je 12 informací (název jednotky, kolik zabírá populace, kolik je potřeba dřeva, železa a akčních bodů k výcviku, síla, bonus síly, obrana, bonus obrany, kolik jednotka unese surovin, kolik políček na mapě ujde za jeden akční bod a poslední z dvanácti vlastností je žold jednotky).

4.15. Kavalerie

Dědí z třídy „Vojak“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.

- Kavalerie
 - * Vstupní parametry:
 - Pocet: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.16. Kopijnik

Dědí z třídy „Vojak“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Kopijnik

* Vstupní parametry:

· Pocet: int

* Vrací: void

* Popis: Jedná se o konstruktor třídy.

4.17. Kostel

Dědí z třídy „Budova“.

- Atributy: Třída má kromě zděděných atributů ještě jeden.

Atribut	Datový typ	Popis
spokojenost	int	Podle hodnoty zděděného atributu „uroven“ nastavuje spokojenost objektu „Kostel“, potažmo „Osada“.

- Vlastnosti: Třída má kromě zděděných Vlastností ještě jednu.

Vlastnost	Datový typ	Čtení/zápis	Popis
Spokojenost	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „spokojenost“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Kostel

* Vstupní parametry:

· uroven: int

* Vrací: void

* Popis: Jedná se o konstruktor třídy.

4.18. Kraj

Třída slouží k objektové manipulaci s jednotlivými políčky na mapě.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
ID_Kraje	int	Identifikační číslo kraje, zároveň udává pozici kraje na mapě.
souradniceX	int	X-ová souřadnice kraje na mapě.
souradniceY	int	Y-ová souřadnice kraje na mapě.
typ	int	Celé číslo od 0 po 4 představující terén kraje, kde 0 a 1 znamená tráva, 2 je les, 3 hory a 4 voda, pro všechny ostatní čísla je terén tma.

- Vlastnosti: Třída má následující Vlastnosti.

Vlastnost	Datový typ	Čtení/zápis	Popis
Pozice	int	Ano/Ne	Vrací hodnotu atributu „ID_Kraje“.
SouradniceX	int	Ano/Ne	Vrací hodnotu atributu „souradniceX“.
SouradniceY	int	Ano/Ne	Vrací hodnotu atributu „souradniceY“.
Typ	int	Ano/Ne	Vrací hodnotu atributu „typ“.

- Metody: Třída má následující metody.

– Kraj

* Vstupní parametry:

- ID_Kraje: int
- SouradniceX: int
- SouradniceY: int
- Typ: int

* Vrací: void

* Popis: Jedná se o konstruktor třídy.

– CtiAdresuObrazkuTypuKraje

* Vrací: string

* Popis: Vrací cestu k souboru s obrázkem příslušného kraje.

4.19. Kraje

Jedná se o statickou třídu. Slouží k zobrazování a přesunu objektů po mapě. Třída má následující statické metody.

- ArmadyVOkoli
 - Vstupní parametry:
 - * stred: Objekt třídy „Kraj“
 - * dim: iteger
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Vrací v poli informace o všech armádách, které jsou ve čtvercovém okolí pozice „stred“ a straně čtverce rovny „dim“.
- NactiKraj
 - Vstupní parametry:
 - * pozice: iteger
 - Vrací: Objekt třídy „Kraj“
 - Popis: Veřejná statická metoda vracející podle pozice objekt třídy „Kraj“.
- OkoliKraje
 - Vstupní parametry:
 - * stred: Objekt třídy „Kraj“
 - * dim: int
 - Vrací: Pole objektů třídy „Kraj“
 - Popis: Veřejná statická metoda. Vrací pole s objekty třídy „Kraj“, které jsou ve čtvercovém okolí pozice „stred“ a straně čtverce rovny „dim“.
- OsadyVOkoli
 - Vstupní parametry:
 - * stred: Objekt třídy „Kraj“
 - * dim: iteger
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Vrací v poli informace o všech osadách, které jsou ve čtvercovém okolí pozice „stred“ a straně čtverce rovny „dim“.

- RadiusPohybuPoKrajich
 - Vstupní parametry:
 - * stred: Objekt třídy „Kraj“
 - * dim: int
 - * radius: int
 - Vrací: Pole objektů třídy „Kraj“
 - Popis: Veřejná statická metoda. Vrací pole s objekty třídy „Kraj“, které jsou ve čtvercovém okolí pozice „stred“ a straně čtverce rovny „dim“ a zároveň vzdálenost těchto krajů od středové pozice je menší nebo rovna hodnotě parametru „radius“. Vzdálenost se počítá jen vodorovně nebo svisle, né šikmo. Kraje sousedící spolu hranou mají vzdálenost rovnu 1, pokud sousedí rohem, tak mají vzdálenost rovnu 2.
- RadiusPohybuPoKrajichDoplnek
 - Vstupní parametry:
 - * stred: Objekt třídy „Kraj“
 - * dim: int
 - * radius: int
 - Vrací: Pole objektů třídy „Kraj“
 - Popis: Veřejná statická metoda. Vrací pole s objekty třídy „Kraj“, které jsou ve čtvercovém okolí pozice „stred“ a straně čtverce rovny „dim“ a zároveň vzdálenost těchto krajů od středové pozice je větší než hodnota parametru „radius“. Vzdálenost se počítá jen vodorovně nebo svisle, né šikmo. Kraje sousedící spolu hranou mají vzdálenost rovnu 1, pokud sousedí rohem, tak mají vzdálenost rovnu 2.
- RozlohaSveta
 - Vrací: Objekt třídy „Rozloha“
 - Popis: Veřejná statická metoda. Vrací objekt třídy „Rozloha“, který reprezentuje rozlohu celého herního světa.
- SpehoveVOkoli
 - Vstupní parametry:
 - * stred: Objekt třídy „Kraj“
 - * dim: iteger
 - * id_hrace: iteger
 - Vrací: ArrayList

- Popis: Veřejná statická metoda. Vrací v poli informace o všech špezech daného hráče, které jsou ve čtvercovém okolí pozice „střed“ a straně čtverce rovny „dim“.

4.20. Lucistnik

Dědí z třídy „Vojak“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Lucistnik
 - * Vstupní parametry:
 - Pocet: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.21. NastaveniTahu

Jedná se o statickou třídu. Třída slouží k provádění dalších tahů hry.

- OsadaTah
 - Vstupní parametry:
 - * $id_{osady} : integer$
 - Vrací: void
 - Popis: Veřejná statická metoda, která provede ve vybrané osadě všechny tahy, které v danou chvíli mají být provedeny.
- HracTah
 - Vstupní parametry:
 - * $id_{hrace} : integer$
 - Vrací: void
 - Popis: Veřejná statická metoda. Na všech objektech patřících danému hráči ověří, zda se má provést tah a pokud ano, tak jej provede.
- KlikNaTah
 - Vstupní parametry:

- * id_hrace: iteger
- Vrací: void
- Popis: Veřejná statická metoda. V databázi u vybraného hráče nastaví v tabulce „Hraci“ v atributu „ClickTah“ na 1. Tato informace pak znamená, že hráč svolil k provedení dalšího tahu celé hry dřív, než uplyne čas vyhrazený pro každý tah.
- MaSeProvetTahHrace
 - Vstupní parametry:
 - * id_hrace: iteger
 - Vrací: void
 - Popis: Sourokomá statická metoda volaná z metody „HracTah“. Vrací rozdíl mezi aktuálním tahem celé hry a posledním tahem hráce.

4.22. Obrance

Dědí z třídy „Armada“.

- Atributy: Třída má kromě zděděných atributů ještě tyto.

Atribut	Datový typ	Popis
mesto	bool	Hodnota udává zda obrance je osada.
hradby	Opevneni	Hodnota udává úroveň budovy opevnění.

- Vlastnosti: Třída má kromě zděděných Vlastností ještě tyto.

Vlastnost	Datový typ	Čtení/zápis	Popis
Mesto	int	Ano/Ne	Vrací hodnotu atributu „mesto“.
Hradby	int	Ano/Ne	Vrací hodnotu atributu „hradby“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Obrance

* Vstupní parametry:

- iD_Obrance: int
- iD_Hrace: int
- mEsto: bool
- hRadby: Objekt třídy „Opevneni“

- Nazev: string
- Pozice: int
- Moralka: int
- Zkusenosti: int
- Sedlak int
- Kopijnik: int
- Sermir: int
- Lucistnik: int
- Povož: int
- Kavalerie: int
- Jidlo: int
- Drevo: int
- Kamen: int
- Zelezo: int
- * Vrací: void
- * Popis: Jedná se o konstruktor třídy.

4.23. Opevneni

Dědí z třídy „Budova“.

- Atributy: Třída má kromě zděděných atributů ještě jeden.

Atribut	Datový typ	Popis
obrana	int	Hodnota tohoto atributu představuje obranný bonus, který se obránci osady připočítává při napadení.

- Vlastnosti: Třída má kromě zděděných Vlastností ještě jednu.

Vlastnost	Datový typ	Čtení/zápis	Popis
Obrana	int	Ano/Ne	Vrací hodnotu atributu „obrana“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Opevneni

- * Vstupní parametry:
 - iD_Obrance: int
 - uroven: int
- * Vrací: void
- * Popis: Jedná se o konstruktor třídy.

4.24. Osada

Třída se skládá z několika budov, které ovlivňují její vlastnosti.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
ID_Osady	int	Hodnota udává identifikační číslo osady.
ID_Hrace	int	Hodnota udává identifikační číslo hráče.
nazev	string	Hodnota udává název osady.
pozice	int	Hodnota udává identifikační číslo kraje nebo-li pozici na mapě.
radnice	Radnice	Hodnota udává radnici v osadě.
kostel	Kostel	Hodnota udává kostel v osadě.
sklad	Sklad	Hodnota udává sklad v osadě.
farma	Farma	Hodnota udává farmu v osadě.
drevorubec	Drevorubec	Hodnota udává dřevorubce v osadě.
kamenolom	Kamenolom	Hodnota udává kamenolom v osadě.
dulNaZezezo	DulNaZezezo	Hodnota udává důl na železo v osadě.
kasarna	Kasarna	Hodnota udává kasárna v osadě.
hrad	Hrad	Hodnota udává hrad v osadě.
opevneni	Opevneni	Hodnota udává opevnění v osadě.
tah	int	Hodnota udává číslo posledního tahu osady.
dane	int	Hodnota udává kolik se každý tah vybere na daních od jednoho obyvatele.

- Vlastnosti: Třída má následující vlastnosti.

Vlastnost	Datový typ	Čtení/zápis	Popis
Id.Osady	int	Ano/Ne	Vrací hodnotu atributu „ID_Osady“.
Id.Hrace	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „ID_Hrace“.
Nazev	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „nazev“.
Pozice	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „pozice“.
Radnice	int	Ano/Ne	Vrací hodnotu atributu „radnice“.
Kostel	int	Ano/Ne	Vrací hodnotu atributu „kostel“.
Sklad	int	Ano/Ne	Vrací hodnotu atributu „sklad“.
Farma	int	Ano/Ne	Vrací hodnotu atributu „farma“.
Drevorubec	int	Ano/Ne	Vrací hodnotu atributu „drevorubec“.
Kamenolom	int	Ano/Ne	Vrací hodnotu atributu „kamenolom“.
DulNaZelezo	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „dulNaZelezo“.
Kasarna	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „kasarna“.
Hrad	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „hrad“.
Opevneni	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „opevneni“.
Tah	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „tah“.
Dane	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „dane“.

- Metody: Třída má následující metody.

- Osada

- * Vstupní parametry:

- iD_Osady: int
 - iD_Hrace: int
 - Nazev: string
 - Pozice: int

- Tah: int
- Pozice: int
- Radnice: Objekt třídy „Radnice“
- Kostel: Objekt třídy „Kostel“
- Sklad: Objekt třídy „Sklad“
- Farma: Objekt třídy „Farma“
- Drevorubec: Objekt třídy „Drevorubec“
- Kamenolom: Objekt třídy „Kamenolom“
- DulNaZezezo: Objekt třídy „DulNaZezezo“
- Kasarna: Objekt třídy „Kasarna“
- Hrad: Objekt třídy „Hrad“
- Opevneni Objekt třídy „Opevneni“
- * Vrací: void
- * Popis: Jedná se o konstruktor třídy.
- CistyPrijem
 - * Vrací: int
 - * Popis: Vrací čistý příjem celé osady.
- HrubyPrijem
 - * Vrací: int
 - * Popis: Vrací hrubý příjem celé osady.
- NakladyNaTezbu
 - * Vrací: int
 - * Popis: Vrací náklady na veškerou těžbu v celé osadě.
- NakladyNaTezbuJidla
 - * Vrací: int
 - * Popis: Vrací náklady na těžbu jídla v celé osadě.
- NakladyNaTezbuDreva
 - * Vrací: int
 - * Popis: Vrací náklady na těžbu dřeva v celé osadě.
- NakladyNaTezbuKamene
 - * Vrací: int
 - * Popis: Vrací náklady na těžbu kamene v celé osadě.
- NakladyNaTezbuZezeza
 - * Vrací: int
 - * Popis: Vrací náklady na těžbu železa v celé osadě.

- MaxDelniku
 - * Vstupní parametry:
 - p: Objekt třídy „Produkce“
 - * Vrací: int
 - * Popis: Vrací maximum dělníků kolik může v produkci pracovat.
- MaxVojaku
 - * Vstupní parametry:
 - v: Objekt třídy „Vojak“
 - * Vrací: int
 - * Popis: Vrací maximum vojáků kolik se může v kasárnách vycvičit.
- MensiZeDvou
 - * Vstupní parametry:
 - a: int
 - b: int
 - * Vrací: int
 - * Popis: Ze dvou celých čísel vrátí to menší.
- NakladyPosadky
 - * Vrací: int
 - * Popis: Vrací žold všech vojáků ubytovaných v kasárnách dané osady.
- NejmensiZ
 - * Vstupní parametry:
 - pole: pole int
 - * Vrací: int
 - * Popis: Z pole celých čísel vrátí to menší.
- PopulaceVProdukci
 - * Vrací: int
 - * Popis: Vrací počet občanů, kteří pracují v kterékoliv z produkcí.
- PrijemJidla
 - * Vrací: int
 - * Popis: Vrací jaký je čistý příjem jídla tzn. těžba jídla minus spotřeba jídla.
- Prirustek
 - * Vrací: int
 - * Popis: Vrací o kolik vzroste nebo se sníží populace.

- SpotrebaJidla
 - * Vrací: int
 - * Popis: Vrací jaká je spotřeba jídla v dané osadě.
- VolnaPopulace
 - * Vrací: int
 - * Popis: Vrací počet občanů, kteří nepracují v žádné z produkcí.

4.25. OsadaInfo

Shromažďuje část informací o osadě. Využití např. k zobrazení seznamu osad, kdy se nemusí načítat z databáze všechny údaje o každé osadě, ale načtou se jen vybraná data.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
Ak	int	Hodnota udává aktuální počet akčních bodů armády.
ID_Hl_Osady	int	Hodnota udává identifikační číslo hlavní osady (tato funkce není ve hře využívána, ale plánuje se její využití v budoucnu).
ID_Hrace	int	Hodnota udává identifikační číslo hráče, kterému osada patří.
ID_Osady	int	Hodnota udává identifikační číslo osady.
JmenoHrace	string	Hodnota udává jméno hráče, kterému osada patří.
NazevOsady	string	Hodnota udává název osady.
Populace	int	Hodnota udává počet občanů v osadě.
Pozice	int	Hodnota udává identifikační číslo kraje (políčka na mapě) na kterém se osada nachází.
PoziceX	int	Hodnota udává ixovou souřadnici pozice.
PoziceY	int	Hodnota udává ypsilonovou souřadnici pozice.
RadniceUroven	int	Hodnota úrovně radnice ve vybrané osadě.

- Vlastnosti: Třída má následující Vlastnosti.

Vlastnost	Datový typ	Čtení/zápis	Popis
ak	int	Ano/Ne	Vrací hodnotu atributu „Ak“.
Id_Hl_Osady	int	Ano/Ne	Vrací hodnotu atributu „ID_Hl_Osady“.
Id_Hrace	int	Ano/Ne	Vrací hodnotu atributu „ID_Hrace“.
Id_Osady	int	Ano/Ne	Vrací hodnotu atributu „ID_Osady“.
jmenoHrace	int	Ano/Ne	Vrací hodnotu atributu „JmenoHrace“.
nazevOsady	int	Ano/Ne	Vrací hodnotu atributu „NazevOsady“.
populace	int	Ano/Ne	Vrací hodnotu atributu „Populace“.
pozice	int	Ano/Ne	Vrací hodnotu atributu „Pozice“.
poziceX	int	Ano/Ne	Vrací hodnotu atributu „PoziceX“.
poziceY	int	Ano/Ne	Vrací hodnotu atributu „PoziceY“.
radniceUroven	int	Ano/Ne	Vrací hodnotu atributu „RadniceUroven“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - OsadaInfo
 - * Popis: Jedná se o konstruktor třídy. Je jednou přetížen. Při první implementaci nepřebírá žádné parametry a hodnoty všech atributů třídy nastavuje na defaultní hodnotu. Při druhé implementaci pak přebírá parametry k nastavení všech atributů třídy.

4.26. Osady

Jedná se o statickou třídu. Slouží k načítání a editaci objektů třídy „Osada“, „Budova“ a „Vojak“. Třída má následující statické metody.

- BudovaZmena
 - Vstupní parametry:
 - * o: Objekt třídy „Osada“
 - * b: Objekt třídy „Budova“
 - Vrací: string

- Popis: Soukromá statická metoda volaná z metod sloužících ke zvýšení úrovně budov. Vrací formátovaný řetězec popisující náklady na zvýšení úrovně budovy
- DrevorubecSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně budovy dřevorubce o jedna.
- DrevorubecZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně budovy dřevorubce o jedna.
- DulNaZelezoSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně dolu na železo o jedna.
- DulNaZelezoZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně dolu na železo o jedna.
- FarmaSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void

- Popis: Veřejná statická metoda. Slouží ke snížení úrovně farmy o jedna.
- FarmaZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně farmy o jedna.
- GenerujOsadu
 - Vstupní parametry:
 - * id_hrace: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda na náhodném místě vytvoří novou osadu s budovami v defaultním nastavení.
- HradSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně hradu o jedna.
- HradZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně hradu o jedna.
- KamenolomSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně kamenolomu o jedna.

- KamenolomZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně kamenolomu o jedna.
- KasarnaSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně kasáren o jedna.
- KasarnaZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně kasáren o jedna.
- KostelSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně kostela o jedna.
- KostelZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně kostela o jedna.
- NactiInfoOOsade

- Vstupní parametry:
 - * id_osady: int
 - Vrací: Objekt třídy „OsadaInfo“
 - Popis: Veřejná statická metoda. Slouží k načtení základních informací o vybrané osadě a z těchto informací vytvoří a vrátí objekt třídy „OsadaInfo“.
- NactiObrance
 - Vstupní parametry:
 - * id_osady: int
 - Vrací: Objekt třídy „Obrance“
 - Popis: Veřejná statická metoda. Slouží k načtení informací o vybrané osadě a z těchto informací vytvoří a vrátí objekt třídy „Obrance“.
 - NactiOsadu
 - Vstupní parametry:
 - * id_osady: int
 - Vrací: Objekt třídy „Osada“
 - Popis: Veřejná statická metoda. Slouží k načtení všech informací o vybrané osadě a z těchto informací vytvoří a vrátí objekt třídy „Osada“.
 - NactiOsady
 - Vstupní parametry:
 - * id_hrace: int
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Metoda vrací pole s objekty třídy „OsadaInfo“ patřící vybranému hráči, pokud hráč žádnou osadu nemá (teprve začíná nebo mu osadu někdo zabral) vygeneruje se mu nová osada.
 - NactiPoziciHLOsady
 - Vstupní parametry:
 - * id_hrace: int
 - Vrací: int
 - Popis: Veřejná statická metoda. Metoda načte a vrátí id hlavní osady.
 - NajdiParcelu

- Vrací: int
- Popis: Soukromá statická metoda volaná z metody „GenerujOsadu“. Metoda ze všech krajů s terénem tráva vybere náhodně jeden a id tohoto kraje vrací.
- NastavDelnikyDrevorubec
 - Vstupní parametry:
 - * id_osady: int
 - * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda v produkci vybrané osady nastaví počet dělníků. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního počtu dělníků v produkci, který je v dané chvíli možný.
- NastavDelnikyDulNaZelezo
 - Vstupní parametry:
 - * id_osady: int
 - * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda v produkci vybrané osady nastaví počet dělníků. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního počtu dělníků v produkci, který je v dané chvíli možný.
- NastavDelnikyFarma
 - Vstupní parametry:
 - * id_osady: int
 - * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda v produkci vybrané osady nastaví počet dělníků. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního počtu dělníků v produkci, který je v dané chvíli možný.
- NastavDelnikyKamenolom
 - Vstupní parametry:

- * id_osady: int
 - * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda v produkci vybrané osady nastaví počet dělníků. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního počtu dělníků v produkci, který je v dané chvíli možný.
- OpevneniSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně opevnění o jedna.
- OpevneniZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně opevnění o jedna.
- PrejmenujOsadu
 - Vstupní parametry:
 - * o: Objekt třídy „Osada“
 - * jmeno: string
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží k přejmenování vybrané osady.
- RadniceSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně radnice o jedna.
- RadniceZvyseni

- Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně radnice o jedna.
- SkladSnizeni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke snížení úrovně skladu o jedna.
 - SkladZvyseni
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží ke zvýšení úrovně skladu o jedna.
 - VycvicKavalerie
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda vycvičí v dané osadě jistý počet kavalerie. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního možného počtu kavalerie v kasárnách, který je v dané chvíli možný.
 - VycvicKopijnik
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - * pocet: int
 - Vrací: void

- Popis: Veřejná statická metoda. Metoda vycvičí v dané osadě jistý počet kopijníků. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního možného počtu kopijníků v kasárnách, který je v dané chvíli možný.
- VycvicLucistnik
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda vycvičí v dané osadě jistý počet lučištníků. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního možného počtu lučištníků v kasárnách, který je v dané chvíli možný.
- VycvicPovoz
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda vycvičí v dané osadě jistý počet povozů. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního možného počtu povozů v kasárnách, který je v dané chvíli možný.
- VycvicSedlak
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“
 - * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda vycvičí v dané osadě jistý počet sedláků. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního možného počtu sedláků v kasárnách, který je v dané chvíli možný.
- VycvicSermir
 - Vstupní parametry:
 - * os: Objekt třídy „Osada“

- * pocet: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda vycvičí v dané osadě jistý počet šermířů. Tento počet se nastavuje podle parametru „pocet“ nebo maximálního možného počtu šermířů v kasárnách, který je v dané chvíli možný.
- VycvicVojaky
 - Vstupní parametry:
 - * o: Objekt třídy „Osada“
 - * v: Objekt třídy „Vojak“
 - Vrací: string
 - Popis: Soukromá statická metoda volaná z metod sloužících k výcviku vojáků. Metoda vrací formátovaný řetězec popisující náklady na výcvik vojáků.
- VytvorKandidatyNaOsadu
 - Vrací: ArrayList
 - Popis: Soukromá statická metoda volaná z metody „NajdiParcelu“. Metoda vrací pole krajů s typem terénu roven tráva.

4.27. Povož

Dědí z třídy „Vojak“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Povož
 - * Vstupní parametry:
 - Pocet: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.28. Produkce

Dědí z třídy „Budova“ a slouží k dědění pro třídy „Farma“, „Drevorubec“, „Kamenolom“ a „DulNaZelezo“.

- Atributy: Třída má kromě zděděných atributů ještě tyto.

Atribut	Datový typ	Popis
delnici	int	Aktuální počet dělníků.
maxDelnici	int	Maximální počet dělníků.
tezba	int	Rychlost těžby (kolik jeden dělník natěží za jeden tah).

- Vlastnosti: Třída má kromě zděděných Vlastností ještě tyto.

Vlastnost	Datový typ	Čtení/zápis	Popis
Delnici	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „delnici“.
MaxDelnici	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „maxDelnici“.
Tezba	int	Ano/Ne	Vrací hodnotu atributu „tezba“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Produkce

- * Popis: Jedná se o konstruktor třídy. Je jednou přetížen. Při první implementaci přebírá parametr „uroven“ a hodnoty všech atributů třídy nastavuje na defaultní hodnotu. Při druhé implementaci pak přebírá navíc parametr „Delnici“ a pomocí těchto dvou nastaví všechny atributy třídy.

4.29. Radnice

Dědí z třídy „Budova“.

- Atributy: Třída má kromě zděděných atributů ještě tyto.

Atribut	Datový typ	Popis
maxAk	int	Podle hodnoty zděděného atributu „uroven“ nastavuje maximum akčních bodů objektu „Radnice“ potažmo „Osada“.
maxPopulace	int	Podle hodnoty zděděného atributu „uroven“ nastavuje maximum populace „Radnice“ potažmo „Osada“.
pocetAk	int	Aktuální počet akčních bodů objektu „Radnice“ potažmo „Osada“.
populace	int	Aktuální počet populace objektu „Radnice“ potažmo „Osada“.

- Vlastnosti: Třída má kromě zděděných vlastnosti ještě tyto.

Vlastnost	Datový typ	Čtení/zápis	Popis
MaxAk	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „maxAk“.
MaxPopulace	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „maxPopulace“.
MaxPopulaceBourani	int	Ano/Ne	Vrací jaká by byla hodnota atributu „maxPopulace“ v případě, že hodnota atributu „uroven“ by byla o jedna menší.
PocetAk	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „pocetAk“.
Populace	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „populace“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Radnice

- * Popis: Jedná se o konstruktor třídy. Je jednou přetížen. Při první implementaci přebírá parametr „uroven“ a hodnoty všech atributů třídy nastavuje na defaultní hodnotu. Při druhé implementaci pak přebírá navíc parametr „Delnici“ a pomocí těchto dvou nastaví všechny atributy třídy.

4.30. Rozloha

Tato třída slouží jako pomocný datový typ, který má dvě celá čísla jen ke čtení.

- Atributy: Třída má kromě zděděných atributů ještě tyto.

Atribut	Datový typ	Popis
velikostX	int	Udává x-ovou velikost herního světa.
velikostY	int	Udává y-ovou velikost herního světa.

- Vlastnosti: Třída má kromě zděděných vlastností ještě tyto.

Vlastnost	Datový typ	Čtení/zápis	Popis
VelikostX	int	Ano/Ne	Vrací hodnotu atributu „velikostX“.
VelikostY	int	Ano/Ne	Vrací hodnotu atributu „velikostY“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Rozloha

- * Popis: Jedná se o konstruktor třídy. Je jednou přetížen. Při první implementaci nepřebírá žádný parametr a hodnoty všech atributů třídy nastavuje na defaultní hodnotu. Při druhé implementaci pak přebírá dva parametry, které nastaví hodnoty obou atributů třídy.

4.31. Sedlak

Dědí z třídy „Vojak“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.

- Sedlak
 - * Vstupní parametry:
 - Pocet: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.32. Sermir

Dědí z třídy „Vojak“.

- Atributy: Třída má jen zděděné atributy.
- Vlastnosti: Třída má jen zděděné Vlastnosti.
- Metody: Třída má jedinou metodu a tou je její konstruktor.

- Sermir
 - * Vstupní parametry:
 - Pocet: int
 - * Vrací: void
 - * Popis: Jedná se o konstruktor třídy.

4.33. Sklad

Dědí z třídy „Budova“.

- Atributy: Třída má kromě zděděných atributů ještě tyto.

Atribut	Datový typ	Popis
jidloSkladem	int	Aktuální množství jídla ve skladě.
drevoSkladem	int	Aktuální množství dřeva ve skladě.
kamenSkladem	int	Aktuální množství kamene ve skladě.
zelezoSkladem	int	Aktuální množství železa ve skladě.
maxSurovin	int	Podle hodnoty zděděného atributu „uroven“ nastavuje maximum skladových prostor.

- Vlastnosti: Třída má kromě zděděných Vlastností ještě tyto.

Vlastnost	Datový typ	Čtení/zápis	Popis
JidloSkladem	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „jidloSkladem“.
DrevoSkladem	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „drevoSkladem“.
KamenSkladem	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „kamenSkladem“.
ZezezoSkladem	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „zezezoSkladem“.
MaxSurovin	int	Ano/Ne	Vrací hodnotu atributu „maxSurovin“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Sklad
 - * Popis: Jedná se o konstruktor třídy. Je jednou přetížen. Při první implementaci přebírá parametr „uroven“ a hodnoty všech atributů třídy nastavuje na defaultní hodnotu. Při druhé implementaci pak přebírá více parametrů pomocí nichž nastaví všechny atributy třídy.

4.34. Speh

Třída dědí od třídy „Jednotka“.

- Atributy: Třída má kromě děděných atributů ještě jeden.

Atribut	Datový typ	Popis
obratnost	int	Hodnota udává obratnost špeha, tato obratnost má vliv na úspěšnost provedení špionáže.

- Vlastnosti: Třída má následující vlastnosti.

Vlastnost	Datový typ	Čtení/zápis	Popis
Obratnost	int	Ano/Ne	Vrací hodnotu atributu „obratnost“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Sklad

- * Popis: Jedná se o konstruktor třídy. Je jednou přetížen. Při první implementaci nepřebírá žádný parametr a hodnoty všech atributů třídy nastavuje na defaultní hodnotu. Při druhé implementaci pak přebírá více parametrů pomocí nichž nastaví všechny atributy třídy.

4.35. SpehInfo

Shromažďuje část informací o špehovi. Využití např. k zobrazení seznamu špehů, kdy se nemusí načítat z databáze všechny údaje o každém špehovi, ale načtou se jen vybraná data.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
Ak	int	Hodnota udává aktuální počet akčních bodů špeha.
ID_Hrace	int	Hodnota udává identifikační číslo hráče, kterému špeh patří.
ID_Osady	int	Hodnota udává identifikační číslo osady v které se špeh nachází.
ID_Speha	int	Hodnota udává identifikační číslo špeha.
JmenoHrace	string	Hodnota udává jméno hráče, kterému špeh patří.
NazevSpeha	string	Hodnota udává jméno špeha.
Obratnost	int	Hodnota udává velikost obratnosti špeha.
Pozice	int	Hodnota udává identifikační číslo kraje (políčka na mapě), na kterém se špeh nachází.
PoziceX	int	Hodnota udává x-ovou souřadnici pozice.
PoziceY	int	Hodnota udává y-ovou souřadnici pozice.
Zkusenosti	int	Hodnota udává počet zkušeností špeha.

- Vlastnosti: Třída má následující Vlastností.

Vlastnost	Datový typ	Čtení/zápis	Popis
ak	int	Ano/Ne	Vrací hodnotu atributu „Ak“.
Id.Hrace	int	Ano/Ne	Vrací hodnotu atributu „ID_Hrace“.
Id.Osady	int	Ano/Ne	Vrací hodnotu atributu „ID_Osady“.
Id.Speha	int	Ano/Ne	Vrací hodnotu atributu „ID_Speha“.
jmenoHrace	int	Ano/Ne	Vrací hodnotu atributu „JmenoHrace“.
nazevSpeha	int	Ano/Ne	Vrací hodnotu atributu „NazevSpeha“.
obratnost	int	Ano/Ne	Vrací hodnotu atributu „Obratnost“.
pozice	int	Ano/Ne	Vrací hodnotu atributu „Pozice“.
poziceX	int	Ano/Ne	Vrací hodnotu atributu „PoziceX“.
poziceY	int	Ano/Ne	Vrací hodnotu atributu „PoziceY“.
zkusenosti	int	Ano/Ne	Vrací hodnotu atributu „Zkusenosti“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - OsadaInfo
 - * Popis: Jedná se o konstruktor třídy. Je jednou přetížen. Při první implementaci nepřebírá žádné parametry a hodnoty všech atributů třídy nastavuje na defaultní hodnotu. Při druhé implementaci pak přebírá parametry k nastavení všech atributů třídy.

4.36. Spehove

Jedná se o statickou třídu. Slouží k načítání a editaci objektů třídy „Speh“ a „SpehInfo“. Třída má následující statické metody.

- ArmadyKeSpionazi
 - Vstupní parametry:
 - * s: Objekt třídy „Speh“
 - Vrací: ArrayList

- Popis: Veřejná statická metoda. Vrací pole s id a jménem všech armád, které může vybraný špeh špehovat. Data se do pole ukládají po dvojicích id a jméno, tzn. v poli s indexem 0 je id první armády, v poli s indexem 1 je jméno první armády atd..
- NactiInfoOSpehu
 - Vstupní parametry:
 - * id_speha: int
 - Vrací: Objekt třídy „SpehInfo“
 - Popis: Veřejná statická metoda. Podle identifikačního čísla špeha načte z databáze data, z nichž vytvoří a vrací objekt třídy „SpehInfo“.
- NactiSpeha
 - Vstupní parametry:
 - * id_speha: int
 - * id_hrace: int
 - Vrací: Objekt třídy „Speh“
 - Popis: Veřejná statická metoda. Podle identifikačního čísla špeha a hráče načte z databáze data, z nichž vytvoří a vrací objekt třídy „Speh“.
- NactiSpehy
 - Vstupní parametry:
 - * id_hrace: int
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Podle identifikačního čísla hráče načte z databáze všechny jeho špehy, vloží je do pole a to vrací.
- OsadaKeSpionazi
 - Vstupní parametry:
 - * s: Objekt třídy „Speh“
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Vrací pole s id a jménem osady, kterou může vybraný špeh špehovat. Pole je buď prázdné (špeh je v místě kde se nenachází žádná osada), nebo má velikost 2 a nese tak informace o jedné osadě.
- Prejmenuj

- Vstupní parametry:
 - * s: Objekt třídy „Speh“
 - * jmeno: string
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží k přejmenování daného špeha.
- Presun
 - Vstupní parametry:
 - * id_Hrace: int
 - * id_Speha: int
 - * kam: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží k přemístění špeha na novou pozici, hodnota nové pozice je v parametru „kam“.
 - UspesnostMise
 - Vstupní parametry:
 - * s: Objekt třídy „Speh“
 - Vrací: int
 - Popis: Veřejná statická metoda. Podle obratnosti špeha rozhodne o úspěšnosti mise, náhodně se vybere číslo od 1 do 100 a porovná se s obratností špeha. Pokud je obratnost větší nebo rovna, mise je úspěšná a vrátí 1. Pokud je obratnost větší nebo rovna jedné třetině vygenerovaného čísla, mise je sice neúspěšná, ale špehovi se podařilo utéct, vrací 2. A pokud je obratnost menší jedné třetině vygenerovaného čísla, špeh byl chycen a zabit (odstraní se z databáze), vrací se 3.
 - VytvorSpeha
 - Vstupní parametry:
 - * o: Objekt třídy „Osada“
 - * jmeno: string
 - Vrací: void
 - Popis: Veřejná statická metoda. Slouží k vytvoření špeha.

4.37. Svet

Jedná se o statickou třídu. Slouží k načítání a vytvoření herního světa.

- AktualniTah
 - Vrací: int
 - Popis: Veřejná statická metoda. Vrací číslo aktuálního čísla tahu celé hry.
- jeKonec
 - Vrací: bool
 - Popis: Veřejná statická metoda. Vrací zda nastal konec hry.
- GenerujSvet
 - Vstupní parametry:
 - * x: int
 - * y: int
 - * lesi: int
 - * hory: int
 - * voda: int
 - Vrací: void
 - Popis: Veřejná statická metoda. Vytvoří herní svět o rozloze x krát y. Parametr „lesi“, stanovuje v procentech rozlohu lesů. Parametr „hory“, stanovuje v procentech rozlohu hor. A parametr „voda“, stanovuje v procentech rozlohu vody.
- PocetHradu
 - Vstupní parametry:
 - * Id_Hrace: int
 - Vrací: int
 - Popis: Veřejná statická metoda. Vrací součet úrovní všech hradů vybraného hráče.

4.38. Vojak

Třída slouží jen k dědění, shromažďuje společné valastnosti všech vojenských jednotek.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
ak1	int	Hodnota udává kolik se vytvoří vojáků za jeden akční bod.
drevo	int	Hodnota udává kolik je potřeba dřeva k výcviku jednoho vojáka.
obrana	int	Hodnota udává obrannou sílu jednoho vojáka.
pocet	int	Hodnota udává počet vojáků daného druhu.
populace	string	Hodnota udává kolik je potřeba populace v osadě k výcviku jednoho vojáka.
rychlost	string	Hodnota udává kolik je potřeba akčních bodů k přesunu vojáka o jedno políčko na mapě.
sila	int	Hodnota udává útočnou sílu jednoho vojáka.
zatez	int	Hodnota udává kolik jeden voják unese surovin (hodnota se počítá od každého druhu, nikoliv všechny suroviny dohromady).
zelezo	int	udává kolik je potřeba železa k výcviku jednoho vojáka.
zold	int	Hodnota udává kolik je žold jednoho vojáka za jeden tah.

- Vlastnosti: Třída má následující Vlastností.

Vlastnost	Datový typ	Čtení/zápis	Popis
Ak1	int	Ano/Ne	Vrací hodnotu atributu „ak1“.
CelkovaObrana	int	Ano/Ne	Vrací obrannou sílu všech vojáků (obrana krát pocet).
CelkovaSila	int	Ano/Ne	Vrací útočnou sílu všech vojáků (sila krát pocet).
CelkovaZatez	int	Ano/Ne	Vrací kolik unesou všichni vojáci (zatez krát pocet).
CelkovyZold	int	Ano/Ne	Vrací žold všech vojáků (zold krát pocet).
Drevo	int	Ano/Ne	Vrací hodnotu atributu „drevo“.
Obrana	int	Ano/Ne	Vrací hodnotu atributu „obrana“.
Pocet	int	Ano/Ano	Vrací a nastavuje hodnotu atributu „pocet“.
Populace	int	Ano/Ne	Vrací hodnotu atributu „populace“.
Rychlost	int	Ano/Ne	Vrací hodnotu atributu „rychlost“.
Sila	int	Ano/Ne	Vrací hodnotu atributu „sila“.
Zatez	int	Ano/Ne	Vrací hodnotu atributu „zatez“.
Zezezo	int	Ano/Ne	Vrací hodnotu atributu „zezezo“.
Zold	int	Ano/Ne	Vrací hodnotu atributu „zold“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.
 - Vojak
 - * Popis: Jedná se o konstruktor třídy. Nepřebírá žádné parametry a hodnoty všech atributů třídy nastavuje na defaultní hodnotu.

4.39. Zprava

Třída představuje zprávu, kterou si hráči mezi sebou posílají.

- Atributy: Třída má následující atributy.

Atribut	Datový typ	Popis
datum	DateTime	Hodnota udává datum a čas vzniku zprávy.
id_Zpravy	int	Hodnota udává identifikační číslo zprávy.
nadpis	string	Hodnota udává nadpis zprávy.
odesilatel	string	Hodnota udává jméno odesílatele.
prijemce	string	Hodnota udává jméno příjemce.
text	string	Hodnota tvoří samotný text zprávy.

- Vlastnosti: Třída má následující Vlastností.

Vlastnost	Datový typ	Čtení/zápis	Popis
datum	DateTime	Ano/Ne	Vrací hodnotu atributu „datum“.
id_Zpravy	int	Ano/Ne	Vrací hodnotu atributu „id_Zpravy“.
nadpis	string	Ano/Ne	Vrací hodnotu atributu „nadpis“.
odesilatel	string	Ano/Ne	Vrací hodnotu atributu „odesilatel“.
prijemce	string	Ano/Ne	Vrací hodnotu atributu „prijemce“.
text	string	Ano/Ne	Vrací hodnotu atributu „text“.

- Metody: Třída má jedinou metodu a tou je její konstruktor.

– Zprava

- * Popis: Jedná se o kostruktor třídy. Je jednou přetížen. Při první implementaci nepřebírá žádné parametry a hodnoty všech atributů třídy nastavuje na defaultní hodnotu. Při druhé implementaci pak přebírá parametry k nastavení všech atributů třídy.

4.40. Zpravy

Jedná se o statickou třídu. Slouží k načítání, vytvoření a mazání zpráv.

- CtiZpravuDorucenou

– Vstupní parametry:

- * id_hrace: int
- * id_zpravy: int

- Vrací: Objekt třídy „Zprava“
- Popis: Veřejná statická metoda. Metoda na základě id hráče a zprávy načte z databáze data a vytvoří z nich objekt třídy „Zprava“, který vrací.
- CtiZpravuOdeslanou
 - Vstupní parametry:
 - * id_hrace: int
 - * id_zpravy: int
 - Vrací: Objekt třídy „Zprava“
 - Popis: Veřejná statická metoda. Metoda na základě id hráče a zprávy načte z databáze data a vytvoří z nich objekt třídy „Zprava“, který vrací.
- Dorucena
 - Vstupní parametry:
 - * id_hrace: int
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Metoda na základě id hráče načte z databáze všechny zprávy doručené danému hráči, vytvoří z nich doručky, vloží je do pole a to vrací.
- Odeslana
 - Vstupní parametry:
 - * id_hrace: int
 - Vrací: ArrayList
 - Popis: Veřejná statická metoda. Metoda na základě id hráče načte z databáze všechny zprávy odeslané daným hráčem, vytvoří z nich doručky, vloží je do pole a to vrací.
- Odeslat
 - Vstupní parametry:
 - * id_hrace: int
 - * komu: string
 - * nadpis: string
 - * text: string
 - Vrací: void

- Popis: Veřejná statická metoda. Metoda odesílá zprávu od hráče, který je zadaný identifikačním číslem, hráči zadaného jménem.
- SmazatDorucene
 - Vstupní parametry:
 - * id_hrace: int
 - * seznamZprav: ArrayList
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda smaže vybrané doručené zprávy daného hráče.
- SmazatOdeslane
 - Vstupní parametry:
 - * id_hrace: int
 - * seznamZprav: ArrayList
 - Vrací: void
 - Popis: Veřejná statická metoda. Metoda smaže vybrané odeslané zprávy daného hráče.

5. Administrace

K administraci je potřeba se dostat na webovou stránku pro přihlášení administrátora. Adresa této stránky je, název domény a za něj napsat „Admin-Log.aspx“. Defaultní přihlašovací údaje jsou, jméno admin, heslo admin. Pokud v databázi neexistuje tabulka „Admin“ přihlášení se nezdaří, ale zároveň se vytvoří nová tabulka „Admin“ s jedním řádkem odpovídajícím defaultním přihlašovacím údajům.

Webové prostředí neumožňuje příliš mnoho zásahů do dění celé hry. Je zde možnost vytvořit databázový systém, opravit procedury, triggerly a pohledy, založit novou hru (což vede k nenávratnému smazání předchozí hry), vytvořit dalšího administrátora a upravit přihlašovací údaje aktuálně přihlášeného administrátora.

Další zásah do dění hry je však možný přímou editací databázových tabulek. Tato editace musí být provedena podle následujících předpisů, jinak by data nebyla konzistentní.

5.1. Přímá editace databáze

V této části se budeme zabývat jen takovými změnami hodnot atributů v tabulkách, které zároveň vyžadují změnu jiných atributů v téže nebo jiných tabulkách.

- Změna pozice armády: Pokud administrátor chce z jakéhokoliv důvodu provést změnu pozice armády, provede to změnou hodnoty atributu „Pozice“ v tabulce „Armáda“. Musí se však nejprve přesvědčit že nová pozice je typu osada nebo tráva, tzn. hodnota atributu „TypKraje“ tabulky „Kraj“ při hodnotě atributu „ID_Kraje“ = číslu nové pozice, je buď „0“ nebo „1“
- Změna pozice špeha: Pokud administrátor chce z jakéhokoliv důvodu provést změnu pozice špeha, provede to změnou hodnoty atributu „Pozice“ v tabulce „Speh“. Musí se však nejprve přesvědčit že nová pozice je typu osada nebo tráva, tzn. hodnota atributu „TypKraje“ tabulky „Kraj“ při hodnotě atributu „ID_Kraje“ = číslu nové pozice, je buď „0“ nebo „1“
- Změna pozice osady: Pokud administrátor chce z jakéhokoliv důvodu provést změnu pozice špeha, provede to změnou hodnoty atributu „Pozice“ v tabulce „Osada“. Musí se však nejprve přesvědčit že nová pozice je typu tráva, tzn. hodnota atributu „TypKraje“ tabulky „Kraj“ při hodnotě atributu „ID_Kraje“ = číslu nové pozice, je „1“. Pokud je podmínka splněna, musí se provést změna i v tabulce „Kraj“ a to sice pro „ID_Kraje“ = původní pozici nastavit hodnotu atributu „TypKraje“ na „1“ a pro „ID_Kraje“ = nové pozici nastavit hodnotu atributu „TypKraje“ na „0“.
- Změna úrovně radnice: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně radnice, musí kromě změny hodnoty atributu „Uroven“ v tabulce „Radnice“, provést i změnu hodnoty atributu „PopulaceMax“ a „AkMax“ (tyto atributy jsou na atributu „Uroven“ funkčně závislé). Hodnoty těchto atributů vypočteme podle vzorce $PopulaceMax = Uroven * 100$, $AkMax = Uroven * Uroven * 2$. Pokud budeme hodnotu atributu „Uroven“ snižovat, musíme dát ještě pozor, aby hodnota atributu „Populace“, nebyla větší než hodnota atributu „PopulaceMax“.
- Změna úrovně kostela: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně kostela, musí kromě změny hodnoty atributu „Uroven“ v tabulce „Kostel“, provést i změnu hodnoty atributu „Spokojenost“ (tento atribut je na atributu „Uroven“ funkčně závislý). Hodnotu tohoto atributu vypočteme podle vzorce $Spokojenost = Uroven * 100$.
- Změna úrovně skladu: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně skladu, musí kromě změny hodnoty atributu „Uroven“ v tabulce „Sklad“, provést i změnu hodnoty atributu „SurovinyMax“ (tento

atribut je na atributu „Uroven“ funkčně závislý). Hodnotu tohoto atributů vypočteme podle vzorce $SurovinyMax = 1000 + (Uroven * Uroven * Uroven * 120)$. Pokud budeme hodnotu atributu „Uroven“ snižovat, musíme dát ještě pozor, aby hodnoty atributů „Jidlo“, „Drevo“, „Kamen“ a „Zezezo“, nebyly větší než je hodnota atributu „SurovinyMax“.

- Změna úrovně farmy: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně farmy, musí kromě změny hodnoty atributu „Uroven“ v tabulce „Farma“, provést i změnu hodnoty atributu „DelniciMax“ (tento atribut je na atributu „Uroven“ funkčně závislý). Hodnotu tohoto atributu vypočteme podle vzorce $DelniciMax = Uroven * 40$). Pokud budeme hodnotu atributu „Uroven“ snižovat, musíme dát ještě pozor, aby hodnota atributu „Delnici“, nebyla větší než je hodnota atributu „DelniciMax“.
- Změna úrovně dřevorubce: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně dřevorubce, musí kromě změny hodnoty atributu „Uroven“ v tabulce „Drevorubec“, provést i změnu hodnoty atributu „DelniciMax“ (tento atribut je na atributu „Uroven“ funkčně závislý). Hodnotu tohoto atributu vypočteme podle vzorce $DelniciMax = Uroven * 40$). Pokud budeme hodnotu atributu „Uroven“ snižovat, musíme dát ještě pozor, aby hodnota atributu „Delnici“, nebyla větší než je hodnota atributu „DelniciMax“.
- Změna úrovně kamenolomu: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně kamenolomu, musí kromě změny hodnoty atributu „Uroven“ v tabulce „Kamenolom“, provést i změnu hodnoty atributu „DelniciMax“ (tento atribut je na atributu „Uroven“ funkčně závislý). Hodnotu tohoto atributu vypočteme podle vzorce $DelniciMax = Uroven * 40$). Pokud budeme hodnotu atributu „Uroven“ snižovat, musíme dát ještě pozor, aby hodnota atributu „Delnici“, nebyla větší než je hodnota atributu „DelniciMax“.
- Změna úrovně dolu na železo: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně dolu na železo, musí kromě změny hodnoty atributu „Uroven“ v tabulce „DulNaZezezo“, provést i změnu hodnoty atributu „DelniciMax“ (tento atribut je na atributu „Uroven“ funkčně závislý). Hodnotu tohoto atributu vypočteme podle vzorce $DelniciMax = Uroven * 40$). Pokud budeme hodnotu atributu „Uroven“ snižovat, musíme dát ještě pozor, aby hodnota atributu „Delnici“, nebyla větší než je hodnota atributu „DelniciMax“.
- Změna úrovně kasáren: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně kasáren, musí kromě změny hodnoty atributu „Uroven“ v tabulce „Kasarna“, provést i změnu hodnoty atributu „MaxVojaci“

(tento atribut je na atributu „Uroven“ funkčně závislý). Hodnotu tohoto atributu vypočteme podle vzorce $\text{MaxVojaci} = \text{Uroven} * 50$). Pokud budeme hodnotu atributu „Uroven“ snižovat, musíme dát ještě pozor, aby součet populace všech jednotek v kasárnách, nebyl větší než je hodnota atributu „MaxVojaci“.

- Změna úrovně opevnění: Pokud administrátor chce z jakéhokoliv důvodu provést změnu úrovně opevnění, musí kromě změny hodnoty atributu „Uroven“ v tabulce „Opevneni“, provést i změnu hodnoty atributu „Obrana“ (tento atribut je na atributu „Uroven“ funkčně závislý). Hodnotu tohoto atributu vypočteme podle vzorce $\text{Obrana} = \text{Uroven} * 5$).

Hodnoty ostatních atributů nejsou již funkčně závislé na jiných a mohou se tak „libovolně“ měnit. To libovolně není úplně přesné, nemůžeme například nastavit hodnotu atributu „ID_Hrace“ v tabulce „Armada“ na id hráče který neexistuje. O to se však stará integritní omezení, takže to ani nejde. Jsou však atributy, které nejsou integritním omezením chráněny, jejich změna mimo logický rozsah však nezpůsobí selhání systému.

5.2. Změna vlastností objektů

Změnu vlastností objektů, provedeme úpravou konstrukturu v třídě, podle kterého se objekt vytváří. Výjimku tvoří vlastnosti, které jsou redundantně uloženy i v databázi. Pokud by jsme u takovýchto vlastností provedli změnu jen v konstruktorech, způsobilo by to, že něco by se provádělo podle starých vlastností a něco podle nových. Musíme tedy provést i editaci databáze. Mezi redundantně uložená data patří následující.

- Objekt „Radnice“: V konstrukturu objektu „Radnice“ je vlastnost „maxPopulace“ a „maxAk“. Hodnota těchto vlastností je uložena i v attributech „PopulaceMax“ a „AkMax“ tabulky „Radnice“.
- Objekt „Kostel“: V konstrukturu objektu „Kostel“ je vlastnost „spokojenost“. Hodnota této vlastnosti je uložena i v atributu „Spokojenost“ tabulky „Kostel“.
- Objekt „Sklad“: V konstrukturu objektu „Sklad“ je vlastnost „maxSurovin“. Hodnota této vlastnosti je uložena i v atributu „SurovinMax“ tabulky „Sklad“.
- Objekt „Farma“: V konstrukturu objektu „Farma“ je vlastnost „tezba“ a V konstrukturu objektu „Produkce“ („Farma“ dědí z „Produkce“) je vlastnost „maxDelnici“. Hodnota těchto vlastností je uložena i v attributech „Tezba“ a „DelniciMax“ tabulky „Farma“.

- Objekt „Drevorubec“: V konstruktoru objektu „Drevorubec“ je vlastnost „tezba“ a V konstruktoru objektu „Produkce“ („Drevorubec“ dědí z „Produkce“) je vlastnost „maxDelnici“. Hodnota těchto vlastností je uložena i v attributech „Tezba“ a „DelniciMax“ tabulky „Drevorubec“.
- Objekt „Kamenolom“: V konstruktoru objektu „Kamenolom“ je vlastnost „tezba“ a V konstruktoru objektu „Produkce“ („Kamenolom“ dědí z „Produkce“) je vlastnost „maxDelnici“. Hodnota těchto vlastností je uložena i v attributech „Tezba“ a „DelniciMax“ tabulky „Kamenolom“.
- Objekt „DulNaZelezo“: V konstruktoru objektu „DulNaZelezo“ je vlastnost „tezba“ a V konstruktoru objektu „Produkce“ („DulNaZelezo“ dědí z „Produkce“) je vlastnost „maxDelnici“. Hodnota těchto vlastností je uložena i v attributech „Tezba“ a „DelniciMax“ tabulky „DulNaZelezo“.
- Objekt „Kasarna“: V konstruktoru objektu „Kasarna“ je vlastnost „maxVojaci“. Hodnota této vlastnosti je uložena i v atributu „MaxVojaci“ tabulky „Kasarna“.
- Objekt „Opevneni“: V konstruktoru objektu „Opevneni“ je vlastnost „obrana“. Hodnota této vlastnosti je uložena i v atributu tabulky „Opevneni“.

Pokud by jsme provedli změnu výpočtu některé z výše zmíněných vlastností objektů, změním i vzorce pro výpočet hodnoty atributů v sekci 5.1..

6. Uživatelská část

Na úvod, jedná se o online tahovou strategii, k hraní vám stačí webový prohlížeč a připojení k internetu. Nejprve je potřeba se zaregistrovat a přihlásit se do hry. Po prvním přihlášení se vám na mapě, která je tvořena čtvercovými políčky, náhodně umístí osada se základními vlastnostmi nezbytnými k dalšímu rozšíření. K provádění jakékoliv činnosti jsou zapotřebí akční body, které se po každém tahu obnovují na svou maximální velikost.

Postupným rozšiřováním počáteční osady se dostanete k výcviku špehů a generálů (tvoří základ armády). Pomocí armády budete schopni útočit na okolní hráče, krást jim suroviny a převážet suroviny mezi svými i cizími osadami. Špehové pak mohou o okolních hráčích zjistit stav surovin, vojáků, úroveň budov a vlastnosti generála.

Pohyb armád a špehů po mapě je omezen počtem akčních bodů, akční body se dají nakoupit výměnou za zkušenosti. Zkušenosti se získávají za každou úspěšně provedenou misi.

Pokud se budete chtít rozšířit o další osadu, máte jedinou možnost jak na to. Musíte drancovat nepřátelskou osadu tak dlouho, dokud jí neklesne budova

radnice na úroveň jedna. Při každém vyhraném útoku na cizí osadu se úroveň všech budov sníží o jedna, maximálně však na úroveň jedna (na nulu to nikdy neklesne). Jakmile je radnice na úrovni jedna a vy vyhrajete bitvu, celá osada je vaše. Má to však jeden háček, tím že se sníží úroveň radnice dobývané osady, sníží se i populace této osady a o tuto sníženou populaci se vám sníží i morálka armády. Při poklesu morálky na nulu se vaše armáda rozprchne a přijdete tak o generála i všechny vojáky ve vaší armádě. Jak zvýšit morálku se dozvíte v sekci armáda.

Kdy nastane tah. Tah nastane nejpozději každý nový den (úderem půlnoci). Druhá možnost je pokud všichni aktivní hráči kliknou na ikonu tah v horním menu. Aktivní hráč je ten, který se za poslední tři tahy alespoň jednou přihlásí do hry, takže po třech dnech bez přihlášení je hráč automaticky neaktivní a ostatní na něj nemusí čekat.

6.1. Finance

Kliknutím na ikonu finance v horním menu se zobrazí přehled příjmů, nákladů a aktuální stav pokladnice. Přehled je rozdělen do tří částí, celkový přehled, přehled v osadách a armádách.

Celkový příjem hráče: Přehled všech příjmů, nákladů a stav pokladnice.

- Příjem osad: Součet příjmů všech osad.
- Náklady armád: Součet nákladů všech armád.
- Čistý příjem: Čistý příjem všech osad.
- Stav pokladnice: Aktuální stav pokladnice.

Přehled financí v osadách: Výpis všech osad hráče s přehledem příjmů a výdajů.

- Název: Název osady.
- Příjem: Hrubý příjem osady.
- Náklady těžby: Náklady těžby v osadě.
- Náklady domobrany: Žold vojáků ubytovaných v kasárnách.
- Celkem: Čistý příjem osady.

Přehled financování armád: Výpis všech armád hráče s přehledem příjmů a výdajů.

- Název: Název armády.
- Náklady: Žold vojáků v armádě.

6.2. Osada

Kliknutím na ikonu osady v horním menu se zobrazí seznam všech osad hráče, po výběru jedné z nich je možno tuto osadu vylepšovat. Každá osada se skládá z deseti budov, které můžeme vylepšovat (dají se i bourat), každá budova slouží k vylepšení vlastností osady. K vylepšení budovy jsou zapotřebí suroviny a akční body, čím vyšší úroveň budovy, tím větší náklady na stavbu.

Důležité je sledovat příjem a spotřebu jídla, s růstem populace roste i spotřeba jídla a pokud natěžíte jídla méně než spotřebují vaši občané, začne se jídlo odečítat ze skladu. Pokud však ve skladě není dostatek jídla, umře vám polovina populace vyhladověním.

Příjem osady je závislý na počtu tzv. volné populace, což je populace osady bez populace pracující v produkci (na farmě, v lesích, v kamenolomu, či v dolech na železo). Čím větší je populace osady, tím více osada vydělává a vy si tak můžete dovolit větší armádu, nebo těžit více surovin a rychleji tak stavět budovy.

Vlastnosti budov:

- Radnice: S každou úrovní budovy radnice roste maximální ubytovací kapacita osady, tj. kolik obyvatel může v osadě žít. Dále je zde přehled o populaci a jejím růstu, akčních bodech, financích, těžbě, velikosti a nákladech domobrany. Tato budova pak ještě slouží i k přejmenování názvu osady.
- Kostel: Tato budova zvyšuje spokojenost obyvatel osady a tím se zvedá růst populace.
- Skladiště: Přehled o stavu surovin v osadě a velikosti skladovací kapacity osady. Čím větší úroveň skladu máte, tím více surovin můžete uskladnit, pokud je sklad plný, suroviny z těžby se nemají kam skladovat a vyhazují se.
- Farma: Slouží k produkci jídla potřebného pro obyvatelstvo osady. Jídlo, které se nepotřebuje se uskladňuje ve skladišti.
- Dřevorubec: Těží se zde dřevo z okolních lesů a uskladňuje ve skladišti.
- Kamenolom: Zajišťuje těžbu kamene, který uskladňuje ve skladišti.
- Důl na železo: Zde se těží železná ruda a zpracováním se z ní vytváří železo, které putuje do skladiště.
- Kasárna: Umožňuje výcvik a ubytování vojáků v osadě. Čím větší úroveň budovy, tím více je možno v kasárnách ubytovat vojáků a zvýšit tak domobranu osady. Výcvik každého vojáka stojí suroviny a akční body.
 - Sedlák: Nejslabší pěší jednotka, výhodou je schopnost unést velkou zátěž.

- Kopijník: Tato jednotka je především k obraně, vyznačuje se velkou obrannou silou, a v případě napadení (jednotka je na straně obránce), se tato obranná síla zvyšuje.
- Šermíř: Je to jednotka, která se díky své velké útočné síle, výborně hodí k útoku. Útočná síla se ještě zvýší, pokud je jednotka na straně útočníka.
- Lučištník: Jednotka má velký význam v první vlně útoku, kdy v řadách útočících pozemních jednotek působí velké ztráty. Druhé vlny útoku se neúčastní.
- Povez: Povezy se bojů prakticky neúčastní a zničeny mohou být jen pokud je armáda totálně poražena. Významnou vlastností je uvést velké množství surovin.
- Kavalerie: V první vlně útoku se chová jako záloha a do bojů nezasahuje, teprve až dojde (jestli dojde) k druhé vlně se při bitvě uplatní její velká síla útoku.

Přehled vlastností těchto jednotek je v prvním rozbalovacím panelu v kasárnách.

Druhý rozbalovací panel pak nabízí možnost výcviku jednotlivých jednotek a zároveň je u každé jednotky zobrazen její počet v osadě a náklady na jejich údržbu.

- Opevnění: V případě napadení zvyšuje obranou sílu bránících jednotek osady.
- Hrad: Slouží k výcviku generálů a špehů. Každá úroveň hradu tak dovolí vycvičit o jednoho generála a o dva špehy navíc. Počítá se součet úrovní všech hradů daného hráče k počtu všech generálů a špehů.

Takže malý příklad, budete mít dvě osady v první hrad na úrovni jedna a v druhé třeba úroveň tři, můžete teby vycvičit $1 + 3$ generály a $2 * (1 + 3)$ špehy.

6.3. Armada

Kliknutím na ikonu armády v horním menu se zobrazí seznam všech armád hráče, po výběru jedné z nich je možno tuto armádu ovládat. Každá armáda se skládá z jednoho generála, podle něhož se i jmenuje a libovolného množství vojenských jednotek (vlastnosti těchto jednotek jsou popsány v sekci *Osada - Kasrna*).

Generál se přímého boje neúčastní, má tři vlastnosti: akční body, morálka a zkušenosti.

- Akční body: Slouží k pohybu po mapě, kdy rychlost celé armády je rovna rychlosti nejpomalejší jednotky. Příklad: budu-li mít v armádě

nejméně jednoho kopijníka (to je nejpomalejší jednotka) a generál bude mít 13 akčních bodů budu moct ujit jen dvě políčka a jeden akční bod by vám zbyl, protože rychlost kopijníka je 6 akčních bodů na jedno políčko mapy.

Akční body můžeme nakupovat výměnou za zkušenosti generála, kdy cena se neustále zvyšuje.

- **Morálka:** Tato vlastnost nesmí být menší nebo rovna nule, jinak se celá armáda rozprchne.

Morálka se sníží za každého zabitého občana nepřátelské osady viz. sekce průběh bitvy.

Zvýšit morálku můžeme nákupem, kdy se za morálku platí zkušenostmi generála.

- **Zkušenosti:** Po vyhrané bitvě se generálovi za každého zabitého nepřátelského vojáka přičtou zkušenosti, které pak může směnit za akční body nebo morálku.

Akce , které může generál provádět.

- **Výměna vojáků:** Výměna vojáků je možná jen s osadami nebo armádami patřícími hráči a nacházejícími se na stejném políčku jako vybraná armáda. Při výměně mezi armádou a osadou je navíc omezení ubytovací kapacity kasáren.

- **Výměna surovin:** Oboustranná výměna surovin je možná jen mezi osadami nebo armádami patřícími stejnému hráči a nacházejícími se na stejném políčku jako vybraná armáda. Cizí osadě nebo armádě můžeme suroviny jen posílat.

Kromě omezení kapacity skladu při výměně mezi armádou a osadou je i omezení při výměně mezi dvěma armádami, kdy zátěž armády je součet zátěže všech jednotek.

- **Útok na nepřítele:** Útok je možno provádět jen na osady nebo armády patřící jinému hráči a nacházející se na stejném políčku jako vybraná armáda.

- **Přejmenování:** Je možné každou armádu přejmenovat, jména mohou být stejná, ale pro lepší orientaci se to nedoporučuje.

- **Skrytí/Odkrytí armády:** Jeli armáda bez vojáků, je možné armádu skryt do některé z vlastních osad. Takto skrytá armáda, se pak na mapě nezobrazuje a nedá se na ni útočit. Pokud však bude osada dobyta a obsazena cizím hráčem, všechny armády ukryté v této osadě budou nenávratně ztraceny.

6.4. Špeh

Kliknutím na ikonu špehové v horním menu se zobrazí seznam všech špehů hráče, po výběru jedné z nich je možné špeha ovládat.

Špeh má tři vlastnosti: akční body, obratnost a zkušenosti.

- **Akční body:** Slouží k pohybu po mapě, kdy rychlost špeha je jeden akční bod za jedno políčko.
Akční body můžeme nakupovat výměnou za zkušenosti, kdy cena se neustále zvyšuje.
- **Obratnost:** Určuje procento úspěšnosti prováděných misí.
Po každé nepovedené misi se obratnost sníží o jedna, maximálně se však sníží na 5. Zvýšit obratnost můžeme nákupem za zkušenosti špeha, cena se neustále zvyšuje a nejvíc lze nakoupit obratnost na 95.
- **Zkušenosti:** Po úspěšně provedené misi se přičtou zkušenostní body.
Při neúspěchu se žádné body nepřičítají.

Akce které může špeh provádět.

Špehovat může armády a osady nepřátel, které jsou na stejném políčku.

- **Špehování armády:**
 - Zjistí velikost armády: Odhalí počty jednotek ve špehované armádě.
 - Zjistí stav surovin: Zjistí kolik surovin armáda nese.
 - Zjistí vlastnosti armády: Vyšpehuje akční body, morálku a zkušenosti nepřátelské armády.
- **Špehování osady:**
 - Zjistí stav surovin: Zjistí kolik surovin se ve skladišti osady nachází.
 - Zjistí stav domobrany: Kolik je vojáků v kasárnách špehované osady.
 - Zjistí úroveň bodov: Zjistí úroveň všech budov nepřítele.

6.5. Mapa

Kliknutím na ikonu mapa v horním menu se zobrazí část mapy světa vycentrovaná na hlavní osadu hráče (to je ta úplně první). Na mapu se můžete dostat i odjinud, v seznamu všech osad, armád a špehů je číslo pozice, odkaz na zobrazení té části mapy, kde se objekt nachází.

Na mapě se zobrazují všechny objekty (osady, armády a špehové), kromě špehů cizích hráčů, ty vidět nejsou. Mapu posouváte šipkami v pravém sloupci, nebo kliknutím na políčko na mapě, kdy se mapa na toto políčko vycentruje. Tlačítka plus a minus v pravém sloupci si můžete zobrazovanou část mapy světa zvětšit či zmenšit.

Při kliknutí na váš objekt se objeví tlačítko zobrazit, kterým se dostanete k ovládání tohoto objektu. Pokud kliknete na vlastní armádu či špeha a poté kliknete na barevné políčko znázorňující trávu, armáda či špeh se na toto políčko přesune (na políčka znázorňující lesy, hory a vodu, se přesunout nedá, stejně tak na políčka, která jsou zešedlá, na ty nemáte dostatek akčních bodů potřebných k přesunu).

Kliknutím na objekt patřící cizímu hráči, je tlačítko pro zobrazení nefunkční, ale je možnost kliknout na jméno hráče a tím se přesunete do psaní nové zprávy právě tomuto hráči.

6.6. Zprávy

Kliknutím na ikonu zprávy v horním menu se zobrazí tři panely: napiš novou zprávu, doručená pošta a odeslaná pošta.

- Napiš novou zprávu: Zde píšete nové zprávy. Vyplníte jméno hráče kterému píšete (pokud hráč neexistuje zpráva se neodešle), krátký nadpis a samotný text zprávy.
- Doručená pošta: Obsahuje všechny doručené zprávy, zobrazují se od nejnovější po nejstarší.
- Odeslaná pošta: Obsahuje všechny odeslané zprávy, zobrazují se od nejnovější po nejstarší.

6.7. Průběh bitvy

K tomu, aby mohla bitva vůbec začít musí mít útočník dostatek síly, celá bitva pak trvá nejdéle dvě kola.

Možné výsledky bitvy:

- Bitva se nekonala: V případě že útočník je příliš slabý se bitva nekoná. Ztráty na obou stranách jsou nulové.
- Vítězem bitvy je obránce: Mohou nastat tři případy
 - Útočník po první vlně útoku přišel o většinu hlavní linie a dal se na ústup.
 - Útočník po druhé vlně útoku přišel o všechny vojáky a armáda byla rozprášena.

- Útočník přežil druhou vlnu útoku, ale utrpěl velké ztráty a dal se na ústup.
- Vítězem bitvy je útočník: Mohou nastat čtyři případy
 - Obránce nemá žádné vojáky.
 - Obránce po první vlně útoku přišel o většinu hlavní linie a kapituloval.
 - Obránce po druhé vlně útoku přišel o všechny vojáky.
 - Obránce přežil druhou vlnu útoku, ale utrpěl velké ztráty a kapituloval.
- Bitva skončila nerozhodně: Na obou stranách je vyrovnaná vojenská síla a nikdo po dvou kolech neutrpěl tak velké ztráty na to, aby musel kapitulovat.

Průběh bitvy:

V první vlně útoku se střetnou jen jednotky tvořeny všemi sedláky, šermíři a polovinou kopijníků proti lučištníkům nepřítele (druhá polovina kopijníků a kavalerie je v záloze, povozy do bitvy nezasahují).

Druhé vlny útoku se lučištníci neúčastní. Střetnou se všichni (teda co přežili první vlnu) sedláci, kopijníci, šermíři a kavalerie.

Pokud tedy chci bitvu rozhodnout v první vlně, musím mít velké množství lučištníků, pokud se však bitva přehoupne do druhé vlny, je kavalerie velice platná jednotka. Kopijníci jsou houževnatí a jsou potřební k udržení bojové linie a přežití obou vln útoku. Šermíři pak díky své síle působí v řadách nepřítele velké ztráty. Sedláci se sice účastní obou vln, ale na výsledek bitvy nemají příliš velký vliv, jsou slabí v obraně i útoku.

7. Závěr

Tato bakalářská práce se věnuje vytvoření online tahové strategie. Takovýchto podobných her je na internetu větší množství, žádná z nich však neumožňuje volný pohyb po herním světě. Celý systém by si zasloužil rozšíření o více objektů a více funkcionalit. Tomuto rozšíření nic nebrání, snad jen dostatek volného času.

Na stránce „<http://drakfdoo.aspone.cz/>“ je provizorně spuštěna tato hra. Stránka administrátora je potom „<http://drakfdoo.aspone.cz/AdminLog.aspx>“. Na adrese „<http://mssql.aspone.cz/freehosting/>“ je pro tuto hru SQL manager, přihlašovací jméno „db2774“ a heslo „admin“.

Grafika použitá v tomto projektu je vlastní produkce, kromě obrázků znázorňující terén na mapě (vystřiženy z hry „Divoké kmeny“) a obrázků s texturou omítky a kamené zídky (staženy z internetu).

Grafické rozhraní využívá i prvky „AjaxControlToolkit“ od společnosti „Microsoft“, zdrojový kód těchto prvků je umístěn v adresáři „Bin“.