

Informatikos fakultetas

**T120B516 Objektinis programų projektavimas**

**Laboratorinių darbų ataskaita**

**Žaidimas „Tower Defence“**

|  |  |  |
| --- | --- | --- |
|  | Studentai: | Mangirdas Kazlauskas, IFF-4/1 Simonas Baltulionis, IFF-4/1  Tautvydas Traška, IFF-4/2 |
| Dėstytojai: | Andrej Ušaniov  Dominykas Barisas |

Kaunas 2017

Turinys

[1. Projekto aprašymas 3](#_Toc494154405)

[2. Laboratorinis darbas Nr. 1 4](#_Toc494154406)

[2.1. Pradinė klasių diagrama 4](#_Toc494154407)

[2.2. Vienintelio objekto (Singleton) projektavimo pavyzdys 5](#_Toc494154408)

[2.2.1. Projektavimo šablono realizacija 5](#_Toc494154409)

[2.2.2. Projektavimo šablono naudojimo pagrindimas 6](#_Toc494154410)

[2.3. Abstraktaus fabriko (Abstract factory) projektavimo pavyzdys (Simonas) 6](#_Toc494154411)

[2.3.1. Projektavimo šablono realizacija 6](#_Toc494154412)

[2.3.2. Projektavimo šablono naudojimo pagrindimas 8](#_Toc494154413)

[2.4. Fabriko metodo (Factory method) projektavimo pavyzdys (Mangirdas) 8](#_Toc494154414)

[2.4.1. Projektavimo šablono realizacija 8](#_Toc494154415)

[2.4.2. Projektavimo šablono naudojimo pagrindimas 10](#_Toc494154416)

[2.5. Strategijos (Strategy) projektavimo pavyzdys (Tautvydas) 10](#_Toc494154417)

[2.5.1. Projektavimo šablono realizacija 10](#_Toc494154418)

[2.5.2. Projektavimo šablono naudojimo pagrindimas 12](#_Toc494154419)

# Projekto aprašymas

Laboratorinių darbų metu kuriamas žaidimas „Tower Defence“. „Tower Defence“ – tai vienas iš strateginių vaizdo žaidimų žanrų, kurio tikslas – apginti tam tikrą teritoriją nuo vis atsirandančių priešų. Gynybai nuo priešų statomi įvairių tipų pastatai – gynybinės struktūros, kurios atakuoja priešus, judančius tam tikru keliu iki turimos apginti teritorijos. Taip pat žaidime egzistuoja įvairių tipų priešai su skirtingais gyvybiniais parametrais, pvz., gyvybės taškais. Vienas iš tokio tipo žaidimų pavyzdžių pateiktas 1 paveikslėlyje.

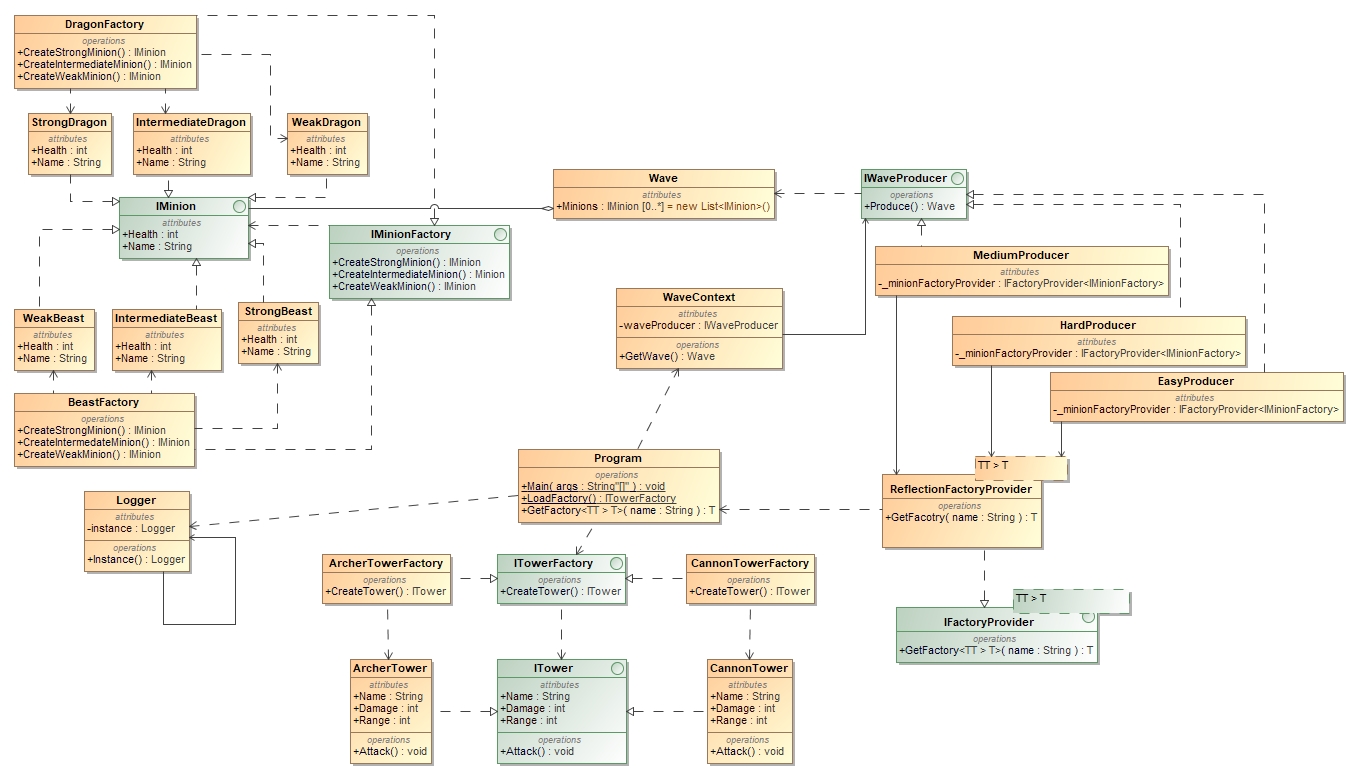


1 pav. Tower Defence tipo žaidimo pavyzdys (šaltinis: <https://i.pinimg.com/originals/02/f2/3f/02f23faf46a076f590a4f4167f365e4c.jpg>)

Tokio tipo žaidimą nutartą kurti dėl to, kad jį labai lengva plėtoti, skirtingi pastatų bei priešų tipai suteikia daug laisvės įvairių programinių struktūrų realizavimui bei dizaino šablonų pritaikymui.

# Laboratorinis darbas Nr. 1

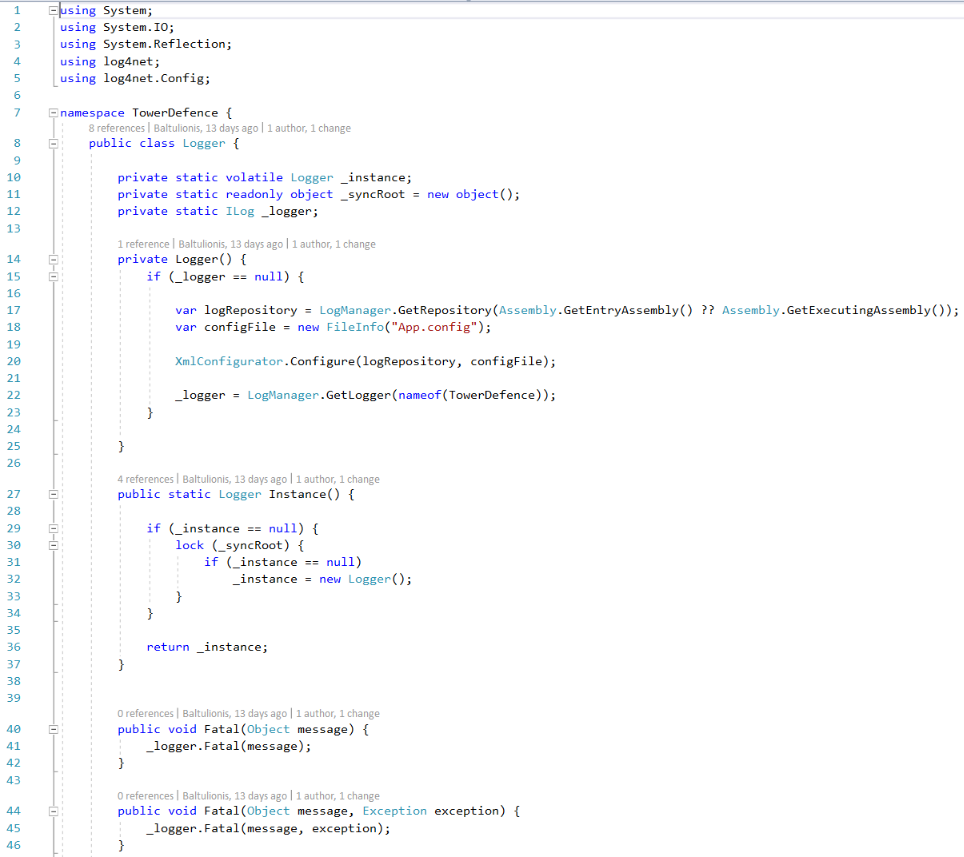
## Pradinė klasių diagrama

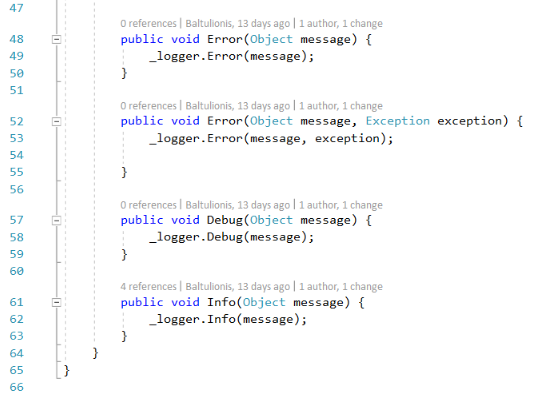
2 paveikslėlyje pateikta pradinė kuriamo žaidimo klasių diagrama.

2 pav. Pradinė klasių diagrama

## Vienintelio objekto (Singleton) projektavimo pavyzdys

### Projektavimo šablono realizacija





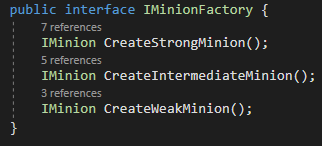
3 pav. Vienintelio objekto (Singleton) realizacija

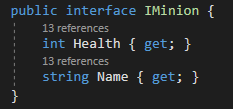
### Projektavimo šablono naudojimo pagrindimas

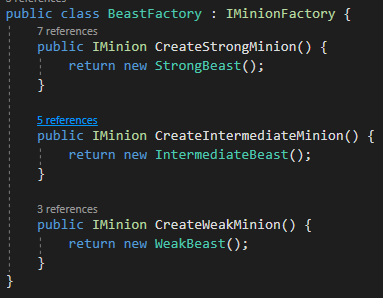
Vienintelio objekto (Singleton) projektavimo šablonas yra naudojamas tada, kai kuri nors klasė projekte yra pastoviai naudojama ir pakartotinai reikia sukurti tos klasės objektą, kas kainuoja resursų. Mūsų atveju tokia klasė – Logger klasė, kuri skirta programos pranešimų išvedimui. Kadangi ši klasė turi daug metodų, savyje talpina nemažai logikos, todėl tokio objekto sukūrimas kas kartą kainuotų nemažai resursų, todėl tikslingai panaudotas Singleton šablonas šią problemą išsprendžia.

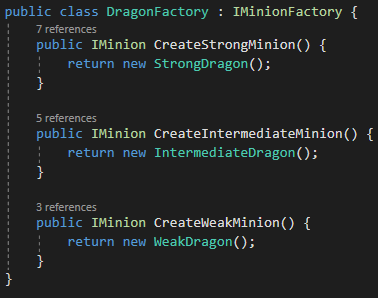
## Abstraktaus fabriko (Abstract factory) projektavimo pavyzdys (Simonas)

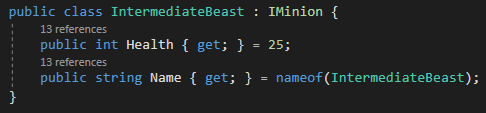
### Projektavimo šablono realizacija

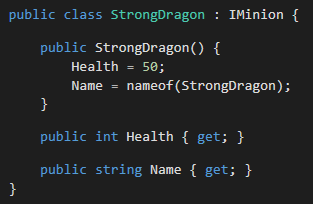












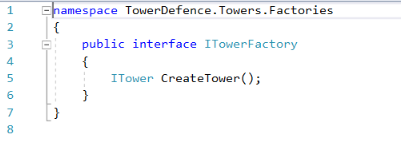
4 pav. Abstraktaus fabriko (Abstract factory) realizacija

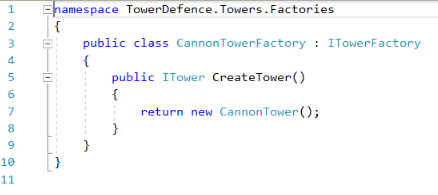
### Projektavimo šablono naudojimo pagrindimas

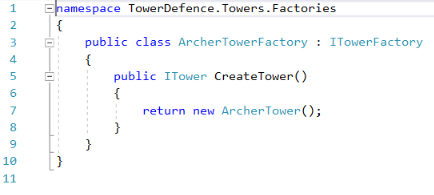
„Abstract factory“ šablonas yra skirtas sukurti objektų šeimoms. Mūsų atveju šablonas naudojamas sukurti priešų objektų šeimoms – galima sukurti trijų tipų žvėris arba tų pačių trijų tipų drakonus. Šie abstraktūs fabrikai gali būti ir bus panaudojami kuriant įvairaus tipo priešų bangas skirtinguose lygiuose.

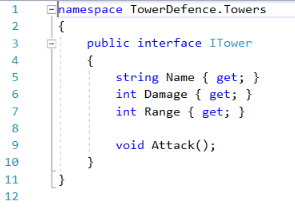
## Fabriko metodo (Factory method) projektavimo pavyzdys (Mangirdas)

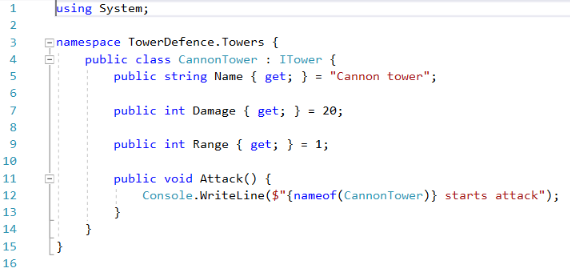
### Projektavimo šablono realizacija

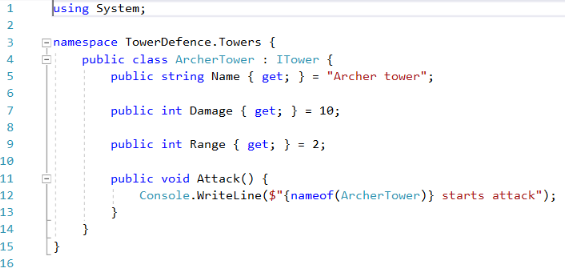


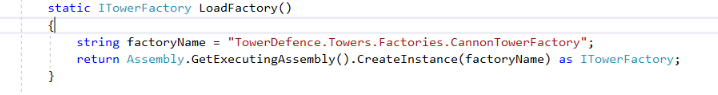












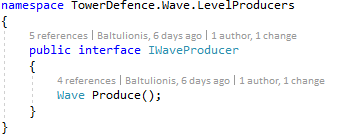
5 pav. Fabriko metodo (Factory method) realizacija

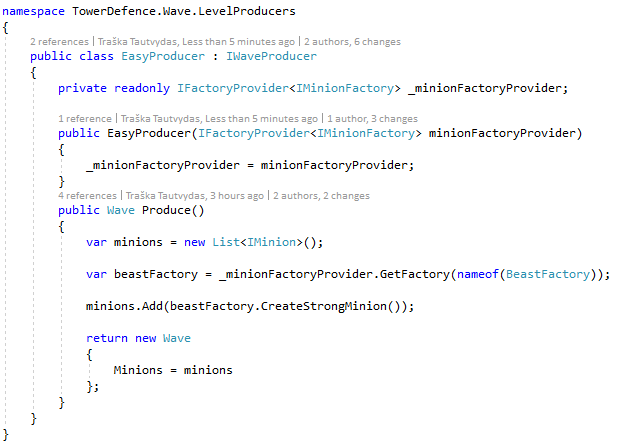
### Projektavimo šablono naudojimo pagrindimas

Fabriko metodo (Factory method) projektavimo šablono paskirtis – palengvinti objektų sukūrimą, viduje paslepiant pačią objekto sukūrimo logiką, o vietoj to naudojant sukurtą sąsają. Taip pat šablonas padeda sukurti objektus, tiksliai neapibrėžiant jų klasės – remiantis šia savybe, buvo sukurtos dvi sąsajos: gynybinių pastatų sąsaja ITower, bei gynybinių pastatų objektų sukūrimo sąsaja ITowerFactory. Naudojantis ITowerFactory sąsaja, galima lengvai sukurti skirtingų tipų fabrikus, kuriančius skirtingų ITower tipų pastatus. Toks projektavimo sprendimas leidžia lengvai plėsti galimų sukurti skirtingų tipų pastatų sąrašą, neapsunkinant ir nekeičiant jau esamų bei naujų pastatų objektų sukūrimo metodų.

## Strategijos (Strategy) projektavimo pavyzdys (Tautvydas)

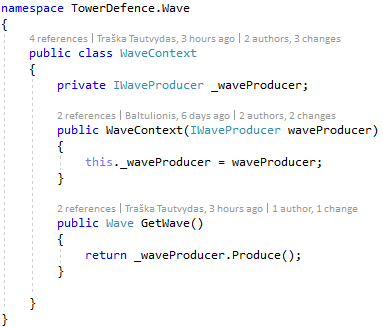
### Projektavimo šablono realizacija











6 pav. Strategijos (Strategy) realizacija

### Projektavimo šablono naudojimo pagrindimas

„Strategy“ šablonas yra skirtas keisti objekto elgseną ar algoritmą programos vykdymo metu. Mūsų atveju šis šablonas naudojamas siekiant parinkti priešų bangų sudėtingumą priklausomai nuo žaidimo lygio. Atsakinga priešų bangos konteksto klasė, objekto kūrimo metu gali priimti skirtingas strategijų implementacijas, lemiančias žaidimo įvairovę ir sudėtingumą.