KAUNO TECHNOLOGIJOS UNIVERSITETAS

INformatikos fakultetas

T120B516 Objektinis programų projektavimas

Projekto ataskaita

Studentų komanda:

Mantas Zambacevičius IFF-4/1,

Ernestas Venckus IFF-4/3,

Žilvinas Abromavičius IFF-4/3

Dėstytojas:

lekt. Barisas Dominykas

KAUNAS, 2017

Turinys

[1 Kuriamo žaidimo aprašymas 3](#_Toc496478111)

[2 Naudojami šablonai 3](#_Toc496478112)

[2.1 Singleton 3](#_Toc496478113)

[2.2 Factory 3](#_Toc496478114)

[2.3 Abstract factory 3](#_Toc496478115)

[2.4 Observer 3](#_Toc496478116)

[2.5 Adapter 3](#_Toc496478117)

[2.6 Prototype 3](#_Toc496478118)

[2.7 Command 3](#_Toc496478119)

[2.8 Bridge 3](#_Toc496478120)

[3 Šablonų esminis kodas 3](#_Toc496478121)

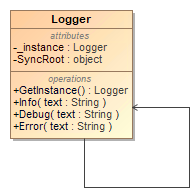
# Kuriamo žaidimo aprašymas

Projekto tikslas – sukurti nedidelį žaidimą, taikant projektavimo šablonus. Kuriamas žaidimas yra 2D šaudyklė.

# Naudojami šablonai

## Singleton

### Klasių diagrama



### Esminis kodas

using System;

namespace Shooter.Classes

{

public class Logger

{

private static volatile Logger \_instance;

private static readonly object SyncRoot = new object();

private Logger() { }

public static Logger Instance

{

get

{

if (\_instance == null)

{

lock (SyncRoot)

{

if(\_instance == null)

\_instance = new Logger();

}

}

return \_instance;

}

}

public void Info(string text)

{

WriteFormattedLog(LogLevel.Info, text);

}

public void Debug(string text)

{

WriteFormattedLog(LogLevel.Debug, text);

}

public void Error(string text)

{

WriteFormattedLog(LogLevel.Error, text);

}

private void WriteFormattedLog(LogLevel level, string text)

{

string pretext;

switch (level)

{

case LogLevel.Info: pretext = "[INFO]"; break;

case LogLevel.Debug: pretext = "[DEBUG]"; break;

case LogLevel.Error: pretext = "[ERROR]"; break;

case LogLevel.Fatal: pretext = "[FATAL]"; break;

default: pretext = ""; break;

}

Console.WriteLine();

Console.WriteLine($"{DateTime.Now} {pretext}{text}");

}

private enum LogLevel

{

Info,

Debug,

Error,

Fatal

}

}

}

### Naudojimo pagrindimas

Logger klasės paskirtis išvesti informacijos, derinimo ir klaidų pranešimus į konsolę. Kadangi visoje programoje reikalingas tik vienas šios klasės objektas, šiai klasėj pritaikomas „Singleton“ dizaino šablonas.

## Factory

## Abstract factory

## Observer

## Adapter

## Prototype

## Command

## Bridge

# Šablonų esminis kodas