

УТВЕРЖДЕНО

Заведующий кафедрой «Управление
разработкой программного обеспечения»
_____ / Авдошин С.М./
« ____ » _____ 2011 г.

**ПРОГРАММА ВИЗУАЛИЗАЦИИ ОПЕРАЦИЙ НАД
КВАТЕРНИОНАМИ НА ПЛАТФОРМЕ WINDOWS PHONE 7**

Программа и методика испытаний

ЛИСТ УТВЕРЖДЕНИЯ

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Руководитель работы

_____ / Гринкруг Е.М./
« ____ » _____ 2011 г.

Исполнитель: студент группы 171ПИ

_____ / Дубов М.С. /
« ____ » _____ 2011 г.

Национальный исследовательский университет – Высшая школа экономики
Факультет бизнес-информатики, отделение программной инженерии

УТВЕРЖДЕНО

**ПРОГРАММА ВИЗУАЛИЗАЦИИ ОПЕРАЦИЙ НАД
КВАТЕРНИОНАМИ НА ПЛАТФОРМЕ WINDOWS PHONE 7**

Программа и методика испытаний

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Листов 9

Содержание

1. Объект испытаний.....	3
1.1. Назначение и область применения.....	3
2. Цели испытаний	3
3. Требования к программе	3
4. Требования к программной документации.....	4
5. Средства и порядок испытаний	4
5.1. Требования к информационной и программной совместимости.....	4
5.2. Требования к составу и параметрам технических средств.....	4
6. Методы испытаний	5
6.1. Испытание развертывания программного комплекса в среде Microsoft Windows Phone 7	5
6.2. Испытание библиотеки DCL.Maths	5
6.3. Испытание библиотеки DCL.Phone.Xna	6
6.4. Испытание приложения Planets.....	6
7. Приложение А. Графические иллюстрации	7
8. Приложение Б. Код автотестов.....	9
8.1. Испытание класса DCL.Maths.Quaternion	9

1. Объект испытаний

Программа визуализации операции над кватернионами на платформе Windows Phone 7 представляет собой программный комплекс, состоящий из трех компонент:

- Первая компонента – библиотека для выполнения основных операций над кватернионами, в дальнейшем именуемая DCL.Maths;
- Вторая компонента – библиотека, содержащая классы, отвечающие за построение и визуализацию трехмерных моделей, их преобразования и повороты посредством операций над кватернионами. Кроме того, библиотека содержит средства для построения пользовательских интерфейсов на платформе Windows Phone 7. В дальнейшем вторая компонента именуется DCL.Phone.Xna;
- Третья компонента – приложение Planets для платформы Windows Phone 7, реализующее визуализацию поворотов трехмерных моделей на примере естественных космических тел.

1.1. Назначение и область применения

1.1.1. Функциональное назначение

Программный комплекс предназначен для визуализации операции поворота трехмерных моделей через кватернионы на платформе Windows Phone 7.

1.1.2. Эксплуатационное назначение

Программа предназначена для эксплуатации пользователем мобильного устройства на платформе Windows Phone 7. Возможна эксплуатация программы в учебном процессе. Предполагается использование библиотек, входящих в состав программы, при разработке других приложений.

2. Цели испытаний

Целью проведения испытаний является проверка корректности работы компонент программного комплекса и правильности комплекта поставки.

3. Требования к программе

В процессе испытаний необходимо проверить соответствие программного комплекса следующим требованиям:

- Программный комплекс корректно развертывается на целевой системе в среде Microsoft Windows Phone при установке приложения из Microsoft Marketplace for Mobile;
- Библиотека DCL.Maths корректно вычисляет результаты операций над кватернионами, со скоростью, обеспечивающей во всех сценах приложения Planets показатель FPS, больший, чем 20;
- Библиотека DCL.Phone.Xna корректно выполняет задачу построения трехмерных моделей и их поворота;
- Библиотека DCL.Phone.Xna обеспечивает построение панорамного пользовательского интерфейса приложения, аналогичного тому, что предоставляет элемент Pivot в Microsoft Silverlight.

- Приложение Planets корректно определяет ось и угол поворота моделей, исходя из анализа манипуляций оператора с помощью средств GUI приложения.

4. Требования к программной документации

В процессе разработки должны быть созданы следующие документы:

- Текст программы (ГОСТ 19.401-78)
- Пояснительная записка (ГОСТ 19.404-79)
- Программа и методика испытаний (ГОСТ 19.301-79)
- Руководство оператора (ГОСТ 19.505-79)
- Руководство программиста (ГОСТ 19.504-79)

5. Средства и порядок испытаний

5.1. Требования к информационной и программной совместимости

5.1.1. Требования к информационной и программной совместимости при запуске программы на устройстве с Windows Phone 7

На устройстве должны быть установлена ОС Windows Phone 7 (или более поздней версии), включающая в себя Microsoft .NET Compact Framework 3.7.

5.1.2. Требования к информационной и программной совместимости при запуске программы на эмуляторе Windows Phone 7

На ПК должна быть установлена интегрированная среда разработки Windows Phone Developer Tools. Работа с Windows Phone Developer Tools возможна в системах Windows® Vista® (x86/x64) с Service Pack 2 и Windows 7 (x86/x64).

5.2. Требования к составу и параметрам технических средств

5.2.1. Требования к составу и параметрам технических средств при запуске программы на устройстве с Windows Phone 7

Необходимый процессор	Рекомендуемый процессор	Необходимое ОЗУ	Рекомендуемое ОЗУ	Необходимое разрешение экрана	Прочие требования
ARM v7 Cortex/Scorpion 1 GHz	ARM v7 Cortex/Scorpion 1 GHz или с более высоким быстродействием	256 MB	256 MB или больше	480x800	GPU с аппаратным ускорением DirectX 9, сенсорный экран

5.2.2. Требования к составу и параметрам технических средств при запуске программы на эмуляторе Windows Phone 7

Необходимый процессор	Рекомендуемый процессор	Необходимое ОЗУ	Рекомендуемое ОЗУ	Необходимое разрешение экрана	Прочие требования
x86/x64 1 GHz	x86/x64 1 GHz или с более высоким быстродействием	2 GB	2 GB или больше	800x600	Графическая карта с драйвером WDDM 1.1, совместимая с DirectX 10, манипулятор типа «мышь»

6. Методы испытаний

6.1. Испытание развертывания программного комплекса в среде Microsoft Windows Phone 7

Программный комплекс распространяется через интернет-магазин приложений Microsoft Marketplace for Mobile.

Развертывание программного комплекса в среде Microsoft Windows Phone осуществляется при помощи следующих шагов:

- Запустить приложение Marketplace на телефоне (либо Zune Software при установке программы с ПК);
- Открыть страницу приложения Planets;
- Нажать кнопку “Buy” и следовать инструкциям на экране (рисунок 1);
- Дождаться появления экрана окончания установки. Приложение Planets появится в списке установленных на телефоне программ;
- Чтобы убедиться в корректности развертывания программного комплекса, запустить приложение и просмотреть его страницы, как описано в руководстве оператора, проверяя, что текстуры на трехмерных моделях соответствуют текстовому содержанию страниц приложения.

6.2. Испытание библиотеки DCL.Maths

Необходимо проверить следующие условия:

- Библиотека считает результаты основных арифметических операций (сложения, умножения) над кватернионами с удовлетворительной точностью;
- Вычисления происходят со скоростью, достаточной, чтобы обеспечить в основном приложении показатель FPS больший, чем 20.

Для проверки первого условия необходимо провести сравнительный анализ результатов вычислений, предоставляемых структурой Quaternion из библиотеки DCL.Maths, с результатами работы аналогичной структуры Quaternion из стандартной библиотеки Microsoft.Xna.Framework. Сравнение осуществляется посредством выполнения автотеста на основе случайной выборки кватернионов (см. приложение 2). Результат работы автотеста:

Successful multiplications: 100%
Successful additions: 100%

– означает, что библиотека DCL.Maths корректно выполняет основные операции над кватернионами.

Для проверки второго условия необходимо воспользоваться компонентом FrameRateCounter, входящим в состав исходного кода программы. Для этого в конструктор Planets() в файле Planets.cs необходимо добавить следующую строку кода:

```
Components.Add(new FrameRateCounter(this));
```

Испытания показывают, что при выполнении программы на телефоне показатель FPS варьируется от 20 до 30, что удовлетворяет требованиям.

6.3. Испытание библиотеки DCL.Phone.Xna

Необходимо проверить следующие условия:

- Библиотека корректно выполняет построение трехмерных моделей, их поворот и вывод на экран;
- Библиотека осуществляет построение пользовательского интерфейса, аналогичного интерфейсу Pivot в Microsoft Silverlight.

Для проверки первого условия необходимо осуществить следующую последовательность действий:

- Запустить приложение Planets на телефоне;
- Убедиться в том, что все представленные в программе модели представляют собой эллипсоиды/сферы
- Убедиться в том, что модели планет вращаются вокруг своих осей с постоянной скоростью, в то время как их спутники вращаются вокруг них по постоянной орбите.

Для проверки второго условия необходимо осуществить следующую последовательность действий:


- Запустить приложение Planets на телефоне;
- Переключиться на соседнюю страницу приложения «перелистывающим» движением пальца в нижней части экрана, как показано на рисунке 2;
- Переключиться на соседнюю страницу приложения нажатием на ее заголовок в верхней части экрана, как показано на рисунке 3;
- Убедиться в том, что проделанные действия аналогичны тем, что требуются для переключения на соседнюю страницу в стандартном Silverlight-приложении Settings.

6.4. Испытание приложения Planets

Необходимо проверить, что приложение корректно определяет ось и угол поворота моделей на основе анализа манипуляций оператора с помощью средств GUI приложения. Для этого следует осуществить следующую последовательность действий:

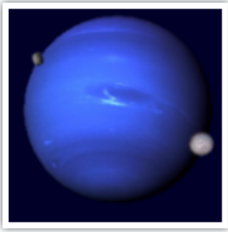
- Запустить приложение Planets на телефоне;
- Провести пальцем в средней части экрана телефона, как показано на рисунке 4;
- Убедиться в том, что поворот моделей осуществляется в направлении движения пальца на угол, зависящий от пройденного пальцем расстояния.

7. Приложение А. Графические иллюстрации


quickplay collection marketplace social

[PICKS](#)
[MUSIC](#)
[VIDEOS](#)
[PODCASTS](#)
[CHANNELS](#)
[APPS](#)

planets



Mikhail Dubov
Entertainment
Version 1.0.0.0
Release date: 5/11/2011

BUY (\$0.99)

FREE TRIAL

Average customer rating:
☆☆☆☆☆

[view reviews](#)

PLANETS

Go off on a virtual journey through our Solar system! With this application you can observe the behaviour of all the 8 planets: Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, Neptune - as well as of their moons; their rotations are modelled in detail. Besides, the application provides the main characteristics for each planet and indicates your position on the globe.




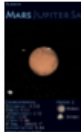

The application has a convenient panoramic interface which enables you to switch the planets by a simple swipe gesture.

FULL VERSION features:

- all the 8 planets with their major moons (only Mercury, Venus, Earth, and Mars are available in the Trial version);
- GeoLocation: indication of the user's position on the model of the Earth.

This app makes use of your phone's:

screenshots








Рис. 1



Рис. 2



Рис. 3



Рис. 4

8. Приложение Б. Код автотестов

8.1. Испытание класса *DCL.Maths.Quaternion*

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using DCL.Maths;

namespace Tests
{
    class Program
    {

        static void Main(string[] args)
        {
            int length = 100;
            DCL.Maths.Quaternion[] dclQuaternions = new DCL.Maths.Quaternion[length];
            Microsoft.Xna.Framework.Quaternion[] xnaQuaternions = new Microsoft.Xna.Framework.Quaternion[length];
            Random rand = new Random();

            for (int i = 0; i < length; i++)
            {
                dclQuaternions[i] = new DCL.Maths.Quaternion((float)rand.NextDouble()*rand.Next(100),
                                                            (float)rand.NextDouble() * rand.Next(100),
                                                            (float)rand.NextDouble()*rand.Next(100),
                                                            (float)rand.NextDouble()*rand.Next(100));
                xnaQuaternions[i] = new Microsoft.Xna.Framework.Quaternion(dclQuaternions[i].X,
                                                                            dclQuaternions[i].Y, dclQuaternions[i].Z, dclQuaternions[i].Re);
            }

            int success = 0;
            DCL.Maths.Quaternion dclTemp;
            Microsoft.Xna.Framework.Quaternion xnaTemp;
            for (int i = 0; i < length - 1; i++)
            {
                dclTemp = dclQuaternions[i] * dclQuaternions[i + 1];
                xnaTemp = xnaQuaternions[i] * xnaQuaternions[i + 1];

                if (dclTemp.Re == xnaTemp.W && dclTemp.X == xnaTemp.X &&
                    dclTemp.Y == xnaTemp.Y && dclTemp.Z == xnaTemp.Z)
                    success++;
            }
            Console.WriteLine("Successful multiplications: " + ((float)success*100/(length-1))+"%");

            success = 0;
            for (int i = 0; i < length - 1; i++)
            {
                dclTemp = dclQuaternions[i] + dclQuaternions[i + 1];
                xnaTemp = xnaQuaternions[i] + xnaQuaternions[i + 1];

                if (dclTemp.Re == xnaTemp.W && dclTemp.X == xnaTemp.X &&
                    dclTemp.Y == xnaTemp.Y && dclTemp.Z == xnaTemp.Z)
                    success++;
            }
            Console.WriteLine("Successful additions: " + ((float)success * 100 / (length - 1)) + "%");
        }
    }
}
```