

Developing Games in Java for Beginners

Saloni Jain

Lecturer in RG PG College, Meerut, U.P, India.

Abstract: The use of JAVA in game development is being discussed in this paper. Games are fast in Darwinism and pliable in every environment. The code designed for games are generally transformable and can be prolonged for a longer duration. Games are generally demarcated by the different versions that are devised by the researchers. Scientists delineates games on the basis of analysing the divergent patterns in the history of a particular game that had come into being. To know the advancements that have taken place in recent times, scholars study the open source projects that highlights the benefits of games.

Keywords: JAVA, Game, GUI

I. INTRODUCTION

Everyone is acquainted with the world of games whether it is played online or video games. This paper gives an insight to all the beginners who are interested to learn about devising games in JAVA. One of the most vital changes that software industry has witnessed is in the field of Games. Games industry have turned out to be one of the most valuable in terms of monetary terms when compared with movie or any other industry. JAVA is being preferred over other languages as it provides users with object-oriented platform which is more efficacious and elementary. Games model provides an interesting research field for the researchers. In today's world the use of technology and especially computers cannot be unnoticed. Games is one the field in which some of the break-through graphics are used. As the games development is very complicated this paper presents easy tricks for the start-up for learners.

II. TECHNIQUE USED

The discrepancy between Games and other software depends on the technique and the expertise that game developers must possess. The prior need is to have a script writer, more technically coder who will design the code. JAVA is object-oriented language. JAVA is a language that doesn't have an icon as it can be more specifically said as a technology. JAVA is preferred over other languages as it is direct accessed for performance and User Interface. Android games are best developed on JAVA.

III. GAME ARCHITECTURE

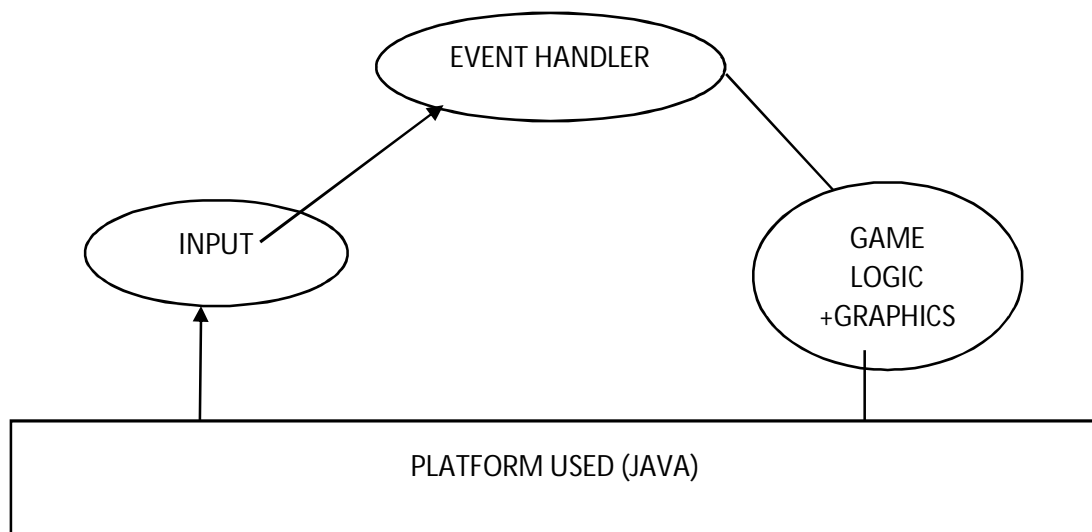


Figure 1: Game Architecture

The above figure shows the platform used in this paper is JAVA. The user gives an input and then the event handler is responsible for performing all the events. It is linked with the logic the game is adapting and the graphics that are being added in the game. The

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

user needs to have a thorough knowledge about JAVA. Mostly, packages used in designing GUI in games are awt(ABSTRACT WINDOW TOOLKIT), applet, swing and util. Awt package is mainly used for designing GUI like buttons, label, text field and other frame for building graphical user interface.

The game developer must first design a window that will incorporate everything that game contains. It can be done as follows:

```
import javax.swing.JFrame;
import javax.swing.WindowConstants;

public class Demo{
    public static void main(String args[]){
        JFrame myFrame = new JFrame(" Frame Demo");
        myFrame.setSize(700,800);
        myFrame.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
        myFrame.setVisible(true);
    }
}
```

Next step is to add animations in the window. Animations are of various kinds like if the developer wants to decorate the window with ball, car and other types of games. This can be added by studying about how to add graphics using the variety of packages offered in JAVA.

If a user desires to add a bouncing ball as an animation then the following strategy can be adopted. The function move() can be coded as:

```
public void move(Container container) {
    x =x+ speedX;
    y =y+ speedY;

    if (x - radius < 0) {

        speedX = -speedX;
        x = radius;
    } else if (x + radius > 1000) {

        speedX = -speedX;
        x = 1000 - radius;
    }

    if (y - radius < 0) {

        speedY = -speedY;
        y = radius;
    } else if (y + radius > 1000) {

        speedY = -speedY;
        y = 1000 - radius;
    }
}
```

This function describes the radius and the speed of the bouncing ball.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

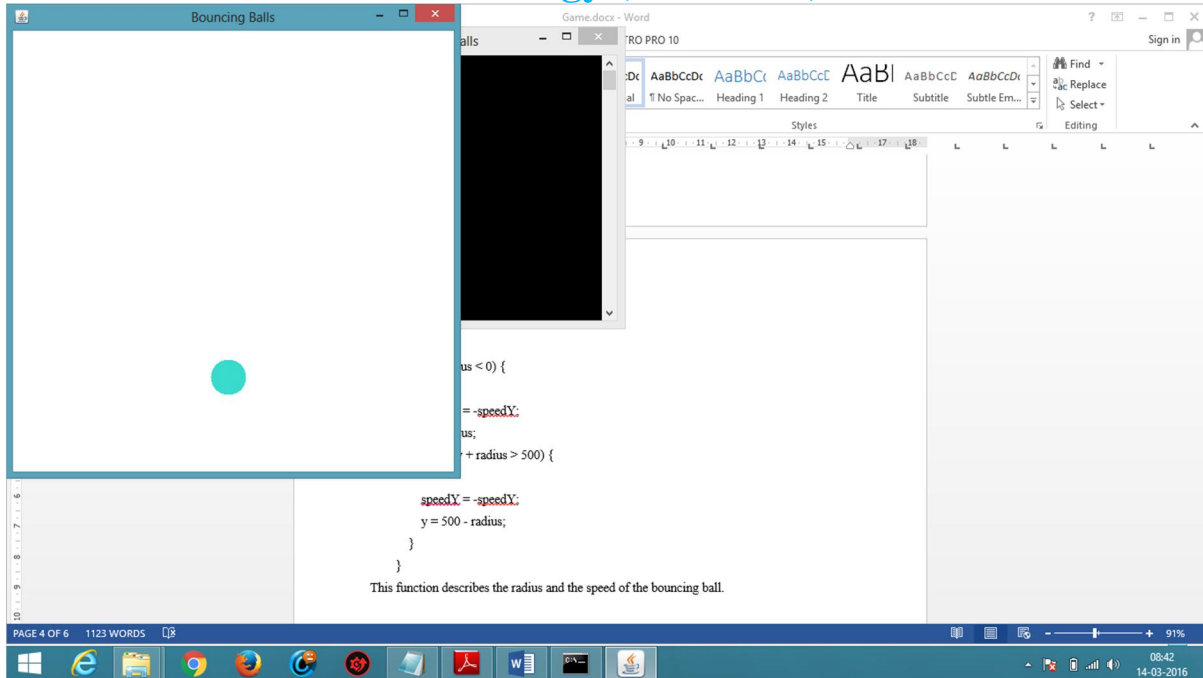


FIGURE 2: Snapshot of the bouncing ball

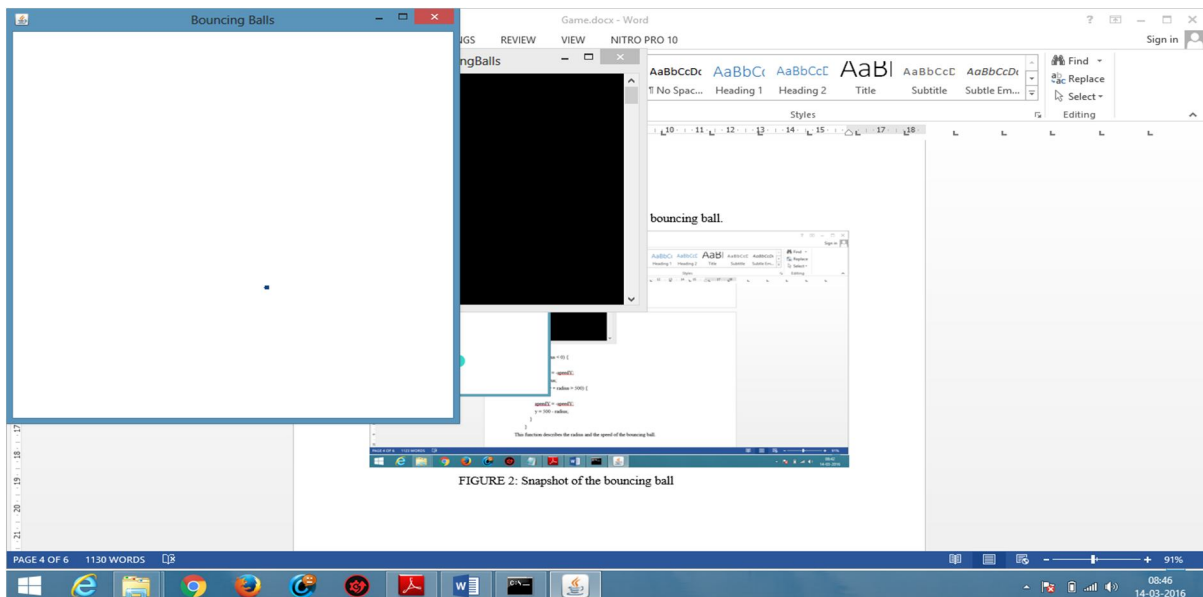


FIGURE 3: Snapshot of ball with smaller radius

IV. CONCLUSION

This paper is aimed at the understanding of easy approach in designing of games in JAVA. Furthermore, as JAVA uses object-oriented technique which is very facile and easy to learn. In order to achieve this goal we examined about the packages used in JAVA for GUI. The various game versions have minimized the intricate and complex model design to a great level. New and newer techniques must be employed and encouraged in the world of game programming.

V. FUTURE SCOPE

This paper discusses about simpler techniques pertaining to games. In future scientists and researchers can extend this by using 3D in developing of games. It will enhance and alter the spectators view regarding games. This can be manipulated by studying the

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

advanced models in this field.

REFERENCES

- [1] C. Alexander, S. Ishikawa, M. Silverstein, A Pattern Language – Town, Buildings, Construction, Oxford University Press, New York, 1977.
- [2] J.M. Bieman, D. Jain, H.J. Yang, OO design patterns, design structure, and program changes: an industrial case study, in: International Conference on Software Maintenance (ICSM 2001), Florence, Italy, November 2001, pp. 580–590.
- [3] L. Bishop, D. Eberly, T. Whitted, M. Finch, M. Shantz, Designing a PC game engine, IEEE Computer Graphics and Application (1998)46–53.
- [4] S. Bjork, S. Lundgren, J. Holopainen, Game design patterns, in: Lecture Note of the Game Design track of Game Developers Conference 2003, March 4–8, San Jose, CA, USA, 2003.
- [5] S. Bjork, S. Lundgren, J. Holopainen, Game design patterns, in: Proceedings of Digital Games Research Conference 2003, Nov. 4–6, Utrecht, The Netherlands, 2003.
- [6] Borland Together Control Center 6.1 product documentation, <http://info.borland.com/techpubs/together/tcc61/>, 2005.
- [7] F. Brito e Abreu, The MOOD Metrics Set, in: Proceedings of the Ninth European Conference Object-Oriented Programming (ECOOP '95) Workshop Metrics, Aug. 1995.
- [8] Aarseth, Espen Cybertext ´ Perspectives on Ergodic Literature. London: Johns Hopkins, 1997.
- [9] Anderson, Craig A. & Dill, Karen E. Video Games and Aggressive Thoughts, Feelings, and Behavior in the Laboratory and in Life. In: Journal of Personality and Social Psychology. Vol 78, no 4, 2000.
- [10] Bartle, Richard. Interactive Multi-User Computer Games. Colchester: MUSE Ltd 1990.
- [11] Steve Rabin, Designing a General Robust AI Engine, In Game Programming Gems, Charles River Media, pp. 221-236, 2000.
- [12] Bjarne Stroustrup (The Creator of C++), Run-Time Type Information, In The C++ Programming Language (Third Edition), Addison Wesley Longman Inc, pp. 407-418, 2000
- [13] Sun Microsystems Inc (Dale Green), The Reflection API, <http://java.sun.com/docs/books/tutorial/reflect/index.html>, 2002
- [14] Trolltech Inc, Qt 3.0 Whitepaper, <http://www.trolltech.com/products/qt/whitepaper/whitepaper.html>, 2002
- [15] Sun Microsystems Inc (Graham Hamilton), JavaBeans, <http://java.sun.com/products/javabeans/docs/spec.html>, 2002
- [16] Sun Microsystems Inc, BeanBox, <http://java.sun.com/docs/books/tutorial/javabeans/beanbox/index.html>, 2002.
- [17] Sun Microsystems Inc, Writing Event Listeners, <http://java.sun.com/docs/books/tutorial/uiswing/events/index.html>, 2002.
- [18] Design Patterns in Smalltalk MVC, Design Patterns, Addison Wesley Longman Inc, pp. 4-6, 1995.