# TETRIS

**A Project Work**

*Submitted in the partial fulfillment for the award of the degree of*

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE ENGINEERING**


**Submitted by:**
**Rishabh Bajpai**

**University Roll Number**

**17BCS4143**

**Under the Supervision of:**
**Mrs. Akwinder**

**SUPERVISORS NAME**
**Gurjot Singh Sodhi**




**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**
**APEX INSTITUE OF TECHNOLOGY**
**CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,**
**PUNJAB**



**MONTH & YEAR**
**October & 2020**

# DECLARATION

I, **'Rishabh Bajpai'**, student of **'Bachelor of Engineering in CSE'**, **session: 2017-2021**, Department of Computer Science and Engineering, Apex Institute of Technology, Chandigarh University, Punjab, hereby declare that the work presented in this Project Work entitled '**Tetris'** is the outcome of our own bona fide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

**(Rishabh Bajpai)**

**CandidateUID:17BCS4143**

# Abstract

A game which is similar to tetris was implemented on an embedded system. The system contained a graphic accelerator which was written in VHDL and a software part written in Python. Using the keyboard as input, two players can play head-to-head until one player looses. The hardware block was intentionally written very generic which made it larger than necessary. This makes it possible to use the same accelerator in other game implementations. The conclusion was drawn that the hardware could be a lot smaller. Using the software to load the memories at startup also doubled the memory requirements. All in all, everything worked as expected but a lot of design improvements is mentioned in the discussion.

# Table of Contents

# 1.INTRODUCTION

The purpose of this project was to build a Tetris video game system using System Verilog and Python language on a FPGA board. Our Tetris game is a single player game where the computer randomly generates tetromino blocks (in the shapes of O, J, L, Z, S, I) that the user can rotate using their game controller. Tetrominoes can be stacked to create lines to be cleared by the computer and be counted as points that will be tracked. Once a tetromino passes the boundary of the screen the user will lose. User input is taken from key inputs from a game controller, and the Tetris sprite based output is displayed using a VGA display. The System Verilog code creates the sprite-based imagery for the VGA display and communicates with the Python language game logic to change what is displayed. Additionally, the System Verilog code generates accompanying audio that will supplement the game in the form of sound effects. To develop this game we required  some specific softwares:

Knowledge of Python

Visual Studio

Unreal engine

Epic game launcher

Windows platform

## 2. PROBLEM FORMULATION

During software development, clones can occur in software intentionally or unintentionally. Developers tend to clone fragments of software during development to save efforts and expedite the development process.

From the literature review, it is observed that studies highlight the need of efficient and scalable approach for detecting code clones having software vulnerability. The existing techniques are not able to detect all types of vulnerable code clones. Different approaches suffer from high false negative rate and not scalable to large software systems due to high time complexity. So firstly, there is a need to optimized out whole setup for this project Second same subject systems should be used to compare the approaches which detect errors.

# 3.RESEARCH OBJECTIVES

The proposed research is aimed to carry out work leading to the development of an approach for vulnerable code clone detection. The proposed aim will be achieved by dividing the work into following objectives:

1. To understand and explore various types of software vulnerabilities existing in open source software.
2. To study and analyse various clone detection techniques that are suitable for vulnerable code clone discovery.
3. To design and develop the technique for vulnerable code clone detection.

To verify and validate the proposed system.

4. To implement individual weapon upgrading system
5. To implement various kinds of enemies that have the purpose to simply attack the player
6. until the death of the player.
7. To add sound effects to player, enemies, weapons and other objects like doors.
8. To design the game levels that will be simple yet beautiful.
9. To make a User Friendly Interface that will be pleasant to look at.
10. To add animations to all sorts of game objects like player, enemies, doors, water
11. including UI transitions.

# 4.METHODOLOGY

The following methodology will be followed to achieve the objectives defined for proposed research work:

1. Detailed study of Python, Epic Game Launcher,Unreal Engine will be done.
2. Installation and hand on experience on existing approaches of Visual Code will be done. Relative pros and cons will be identified.
3. Various parameters will be identified to evaluate the proposed system.
4. Comparison of new implemented approaches with existing approaches will be done.

## 5.RESULTS AND DISCUSSION

More entertaining and more graphics can be added in future. It is forecasted that the gaming industry will generate great opportunities for **game** developers over the next two years. The gaming industry has grown its presence over various platforms – mobile, console, PC, online gaming and the industry is growing really fast along with the presence on all these platforms.

# 6. REFERENCES

C. Anderson (2007). The Innovation in Gaming Isn't on the Screen.

R.T. Beckwith, L. Brandt, B.M. Slator (2006). Electric Worlds in the Classroom: Teaching And Learning With Role-Based Computer Games. Teachers College Press. 182 p.

D.A. Bowman, C. North, J. Chen, N.F. Polys, P.S. Pyla, and U. Yilmaz (2003). Information-Rich Virtual Environments: Theory, Tools, and Research Agenda. Proceedings of the ACM symposium on Virtual reality software and technology, Osaka, Japan, 2003.

J. S. Brown, A. Collins and P. Duguid 1989. Situated Cognition and the Culture of Learning. Educational Researcher, 18(1), 32-42.

The Daedalus Project.
Online MMORPG research publication.

T. M. Duffy, J. Lowyck and D. H. Jonassen, 1983. Designing Environments for Constructive Learning. New York: Springer-Verlag

Game Research
From their website: "Game Research attempts to bring together knowledge on computer games from the areas of art, business, and science. Traditionally such cross-communication has been sparse to the detriment of all involved."