Rock-Paper-Scissors Game (Python Tkinter)

This PDF contains the source code of a Rock-Paper-Scissors game built with Python Tkinter. The game allows a player to play against the computer with a countdown before each round. Both player and computer start with 3 lives, and the game continues until one runs out of lives.

```
import tkinter as tk
import random
import threading
import time

# --- Game State ---
player_lives = 3
computer_lives = 3
countdown = 3

# --- Game Logic ---
def determine_winner(player, computer):
    if player == computer:
        return "tie"
    elif (player == "rock" and computer == "scissors") or          (player == "paper" and computer == "r
        return "player"
    else:
        return "computer"

def update_lives(winner):
    global player_lives, computer_lives
    if winner == "player":
        computer_lives -= 1
    elif winner == "computer":
        player_lives -= 1

def reset_game():
    global player_lives, computer_lives
    player_lives = 3
    computer_lives = 3
    result_label.config(text="")
    lives_label.config(text=f"You ❤■ {player_lives}  |  Computer ❤■ {computer_lives}")
    btn_frame.pack()
    restart_btn.pack_forget()

def play_round(player_choice):
    btn_frame.pack_forget()  # Disable buttons during countdown
    countdown_label.config(text="Get ready...")

    def countdown_timer():
        global player_lives, computer_lives
        for i in range(countdown, 0, -1):
            countdown_label.config(text=f"{i}...")
            time.sleep(1)

        computer_choice = random.choice(["rock", "paper", "scissors"])
        winner = determine_winner(player_choice, computer_choice)
        update_lives(winner)

        # Display outcome
        if winner == "tie":
            result_text = f"Both chose {player_choice} ■ It's a tie!"
        elif winner == "player":
            result_text = f"You chose {player_choice}, computer chose {computer_choice} ■ You win this r
        else:
            result_text = f"You chose {player_choice}, computer chose {computer_choice} ■ Computer wins

        result_label.config(text=result_text)
        lives_label.config(text=f"You ❤■ {player_lives}  |  Computer ❤■ {computer_lives}")
        countdown_label.config(text="")

        if player_lives == 0:
            result_label.config(text="■ Computer wins the game! Better luck next time.")
            restart_btn.pack()
        elif computer_lives == 0:
            result_label.config(text="■ You won the game! Congrats!")
            restart_btn.pack()
        else:
            btn_frame.pack()

    threading.Thread(target=countdown_timer).start()

# --- GUI Setup ---
root = tk.Tk()
root.title("Rock-Paper-Scissors Showdown")
root.geometry("400x400")
root.config(bg="lavender")

title = tk.Label(root, text="Rock-Paper-Scissors", font=("Arial", 20, "bold"), bg="lavender")
title.pack(pady=10)

lives_label = tk.Label(root, text=f"You ❤■ {player_lives}  |  Computer ❤■ {computer_lives}", font=("Ar
lives_label.pack(pady=5)

countdown_label = tk.Label(root, text="", font=("Arial", 18), fg="tomato", bg="lavender")
countdown_label.pack(pady=10)

btn_frame = tk.Frame(root, bg="lavender")
btn_frame.pack(pady=10)

tk.Button(btn_frame, text="■ Rock", width=10, command=lambda: play_round("rock")).grid(row=0, column=0,
tk.Button(btn_frame, text="■ Paper", width=10, command=lambda: play_round("paper")).grid(row=0, column=1
tk.Button(btn_frame, text="✌ Scissors", width=10, command=lambda: play_round("scissors")).grid(row=0, co

result_label = tk.Label(root, text="", font=("Arial", 14), wraplength=350, bg="lavender")
result_label.pack(pady=20)

restart_btn = tk.Button(root, text="Play Again", command=reset_game, bg="lightgray")

root.mainloop()
```