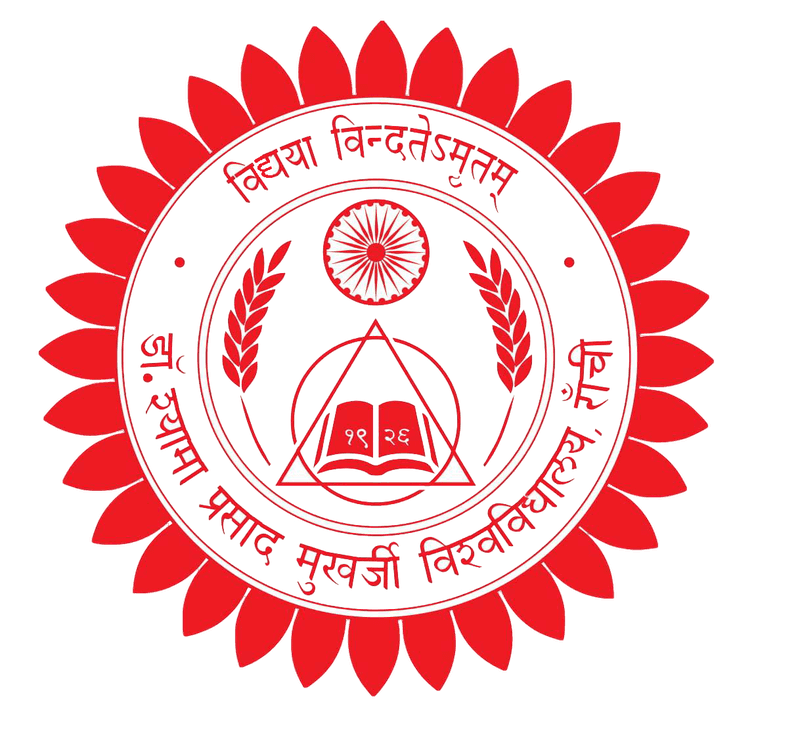
Dr. Shyama Prasad Mukherjee University

****

**DECISION REVIEW SYSTEM**

***A project submitted in partial fulfilment of the requirement for the award of the degree of***

**Bachelor of Science**

**In**

**INFORMATION TECHNOLOGY**

**Submitted By:**

**Name: VISHAL KUMAR**

**Class Roll No: 108**

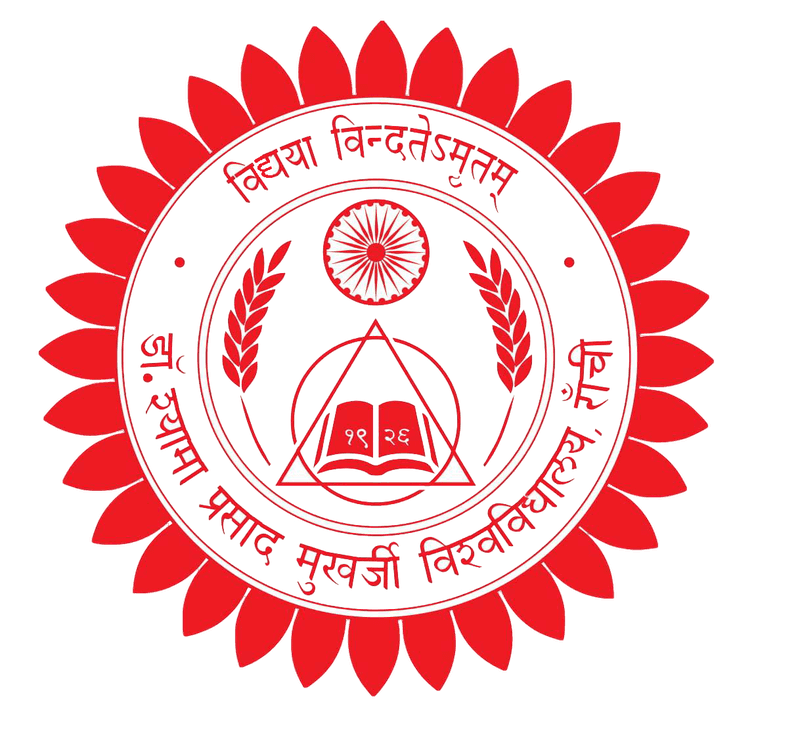
**Exam Roll No: 20B521603**

**Reg. Roll No: DSPMU2020002095**

**Department: B.Sc. IT, Sem: VI**

**Session: 2020-2023**

**DR. SHYAMA PRASAD MUKHERJEE UNIVERSITY MORABADI, RANCHI -834008**



**CERTIFICATE FOR PROJECT**

This is to certify that this is a bonafide record of the project work entitled **“Decision Review System”** done satisfactory at “DSPMU, Ranchi**”** by

**Vishal Kumar,** in partial fulfillment of B.sc (IT) Examination.

This report or similar report on the topic has not been submitted for any other examination and doesn’t form part of any other course undergone by the candidate.

Anchal Kumari

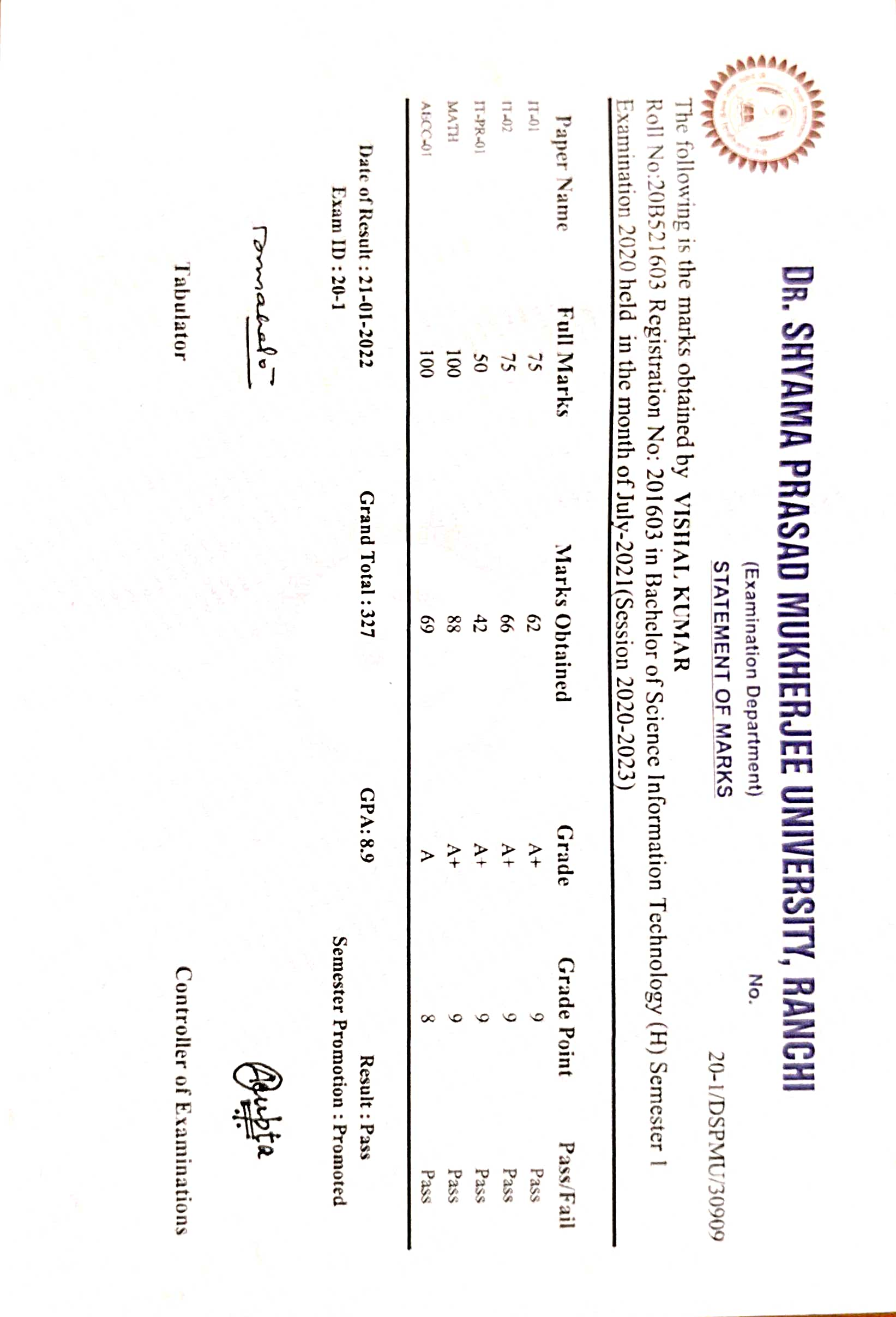
**INTERNAL GUIDE EXTERNAL GUIDE**

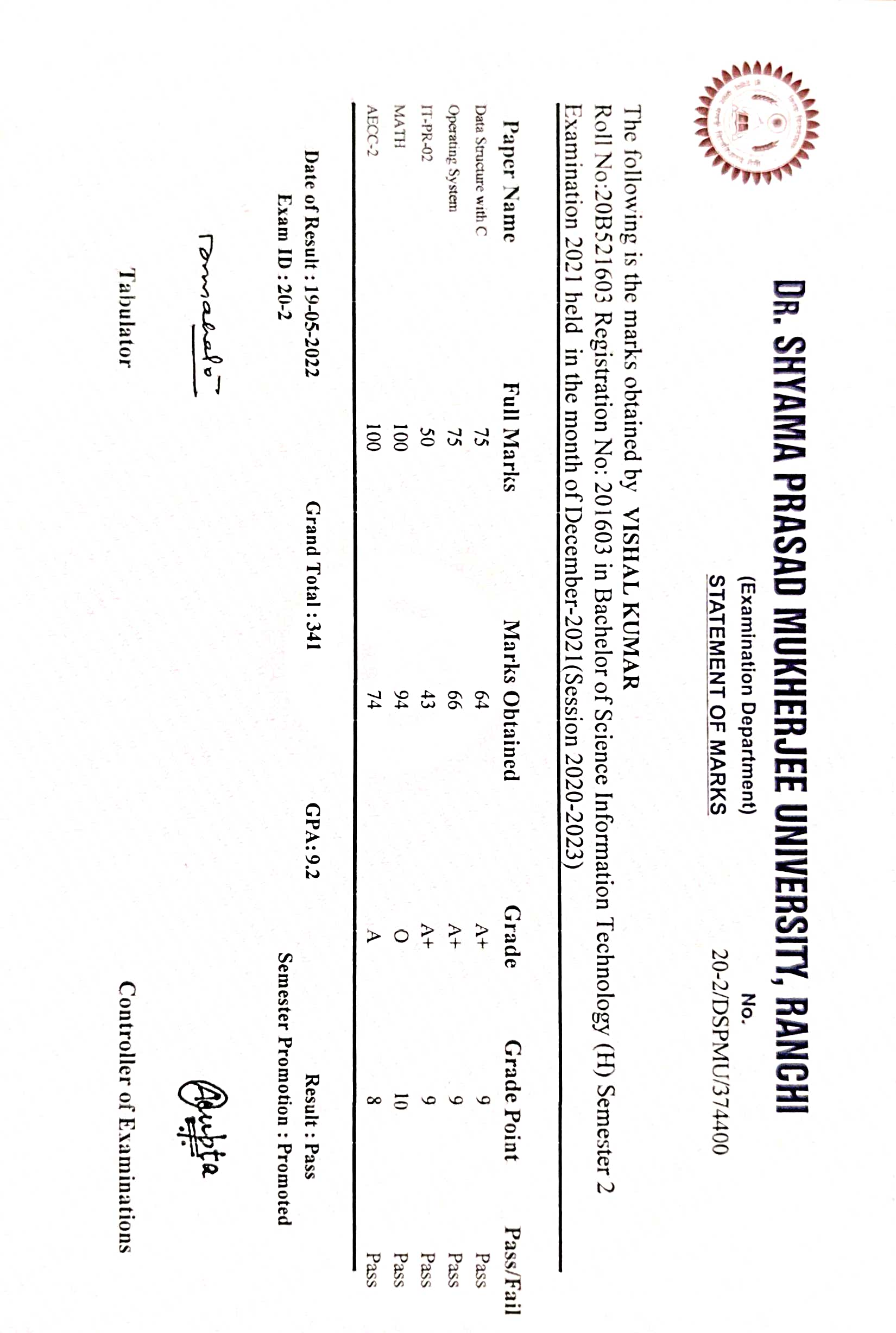
**CERTIFICATE OF OJT 1**

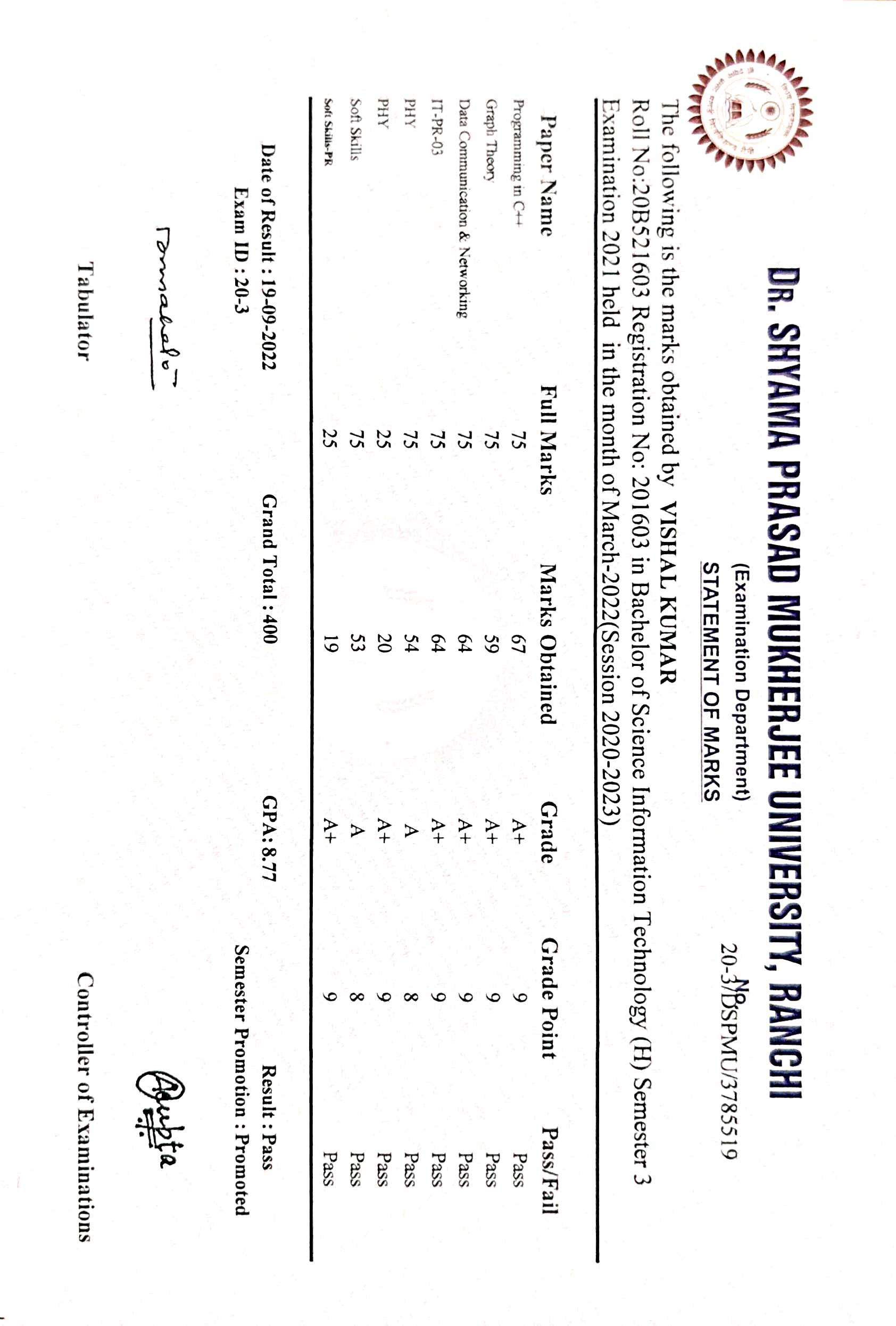


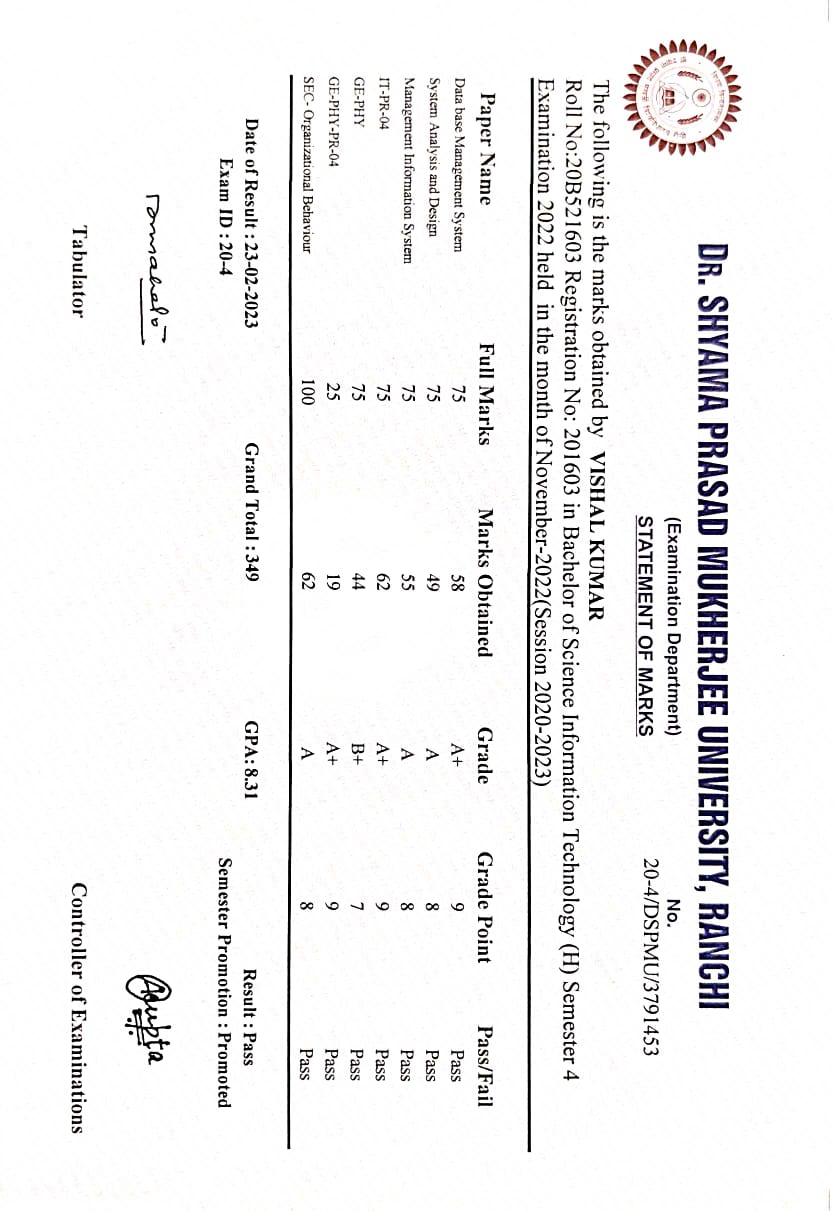
**CERTIFICATE OF OJT 2**

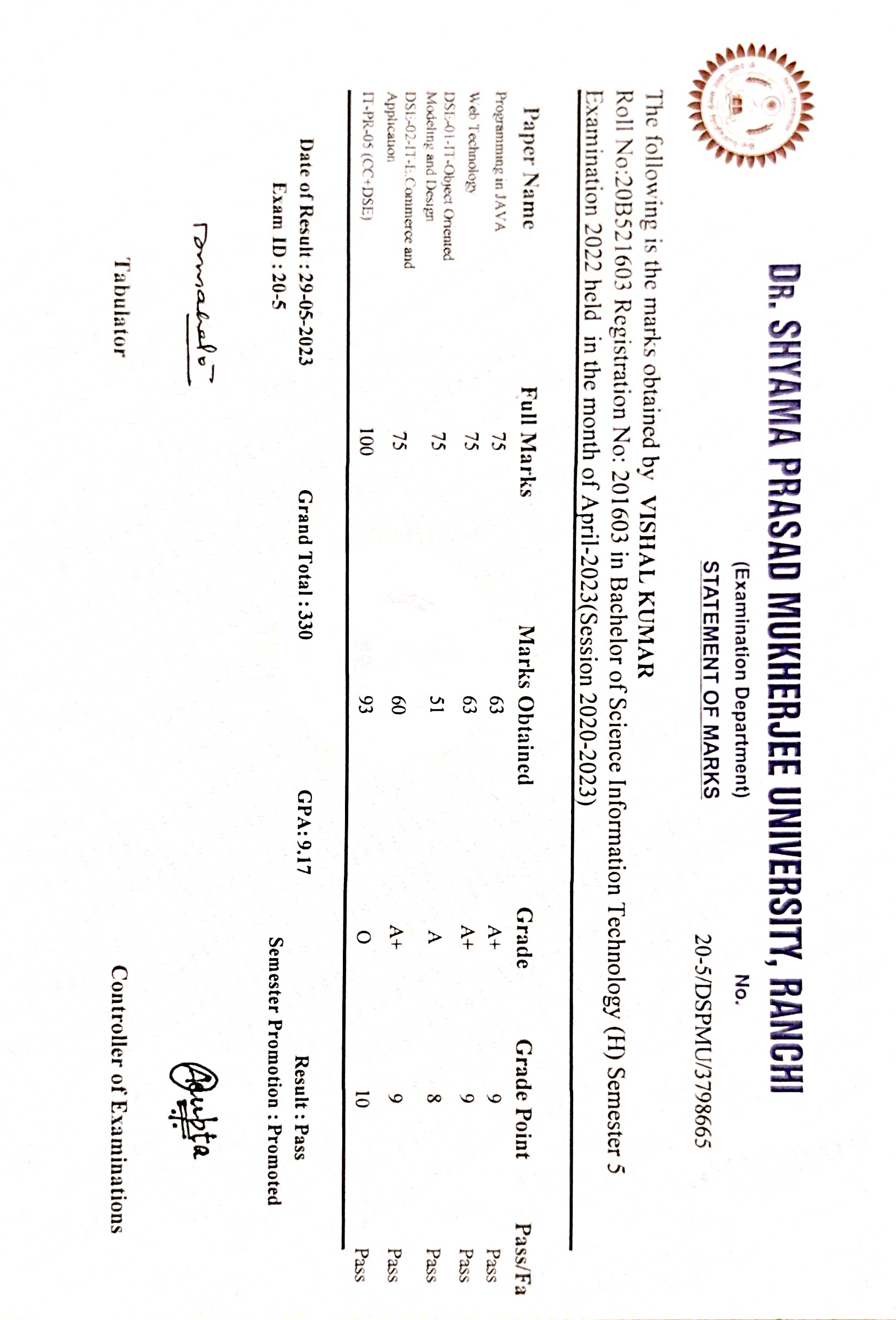
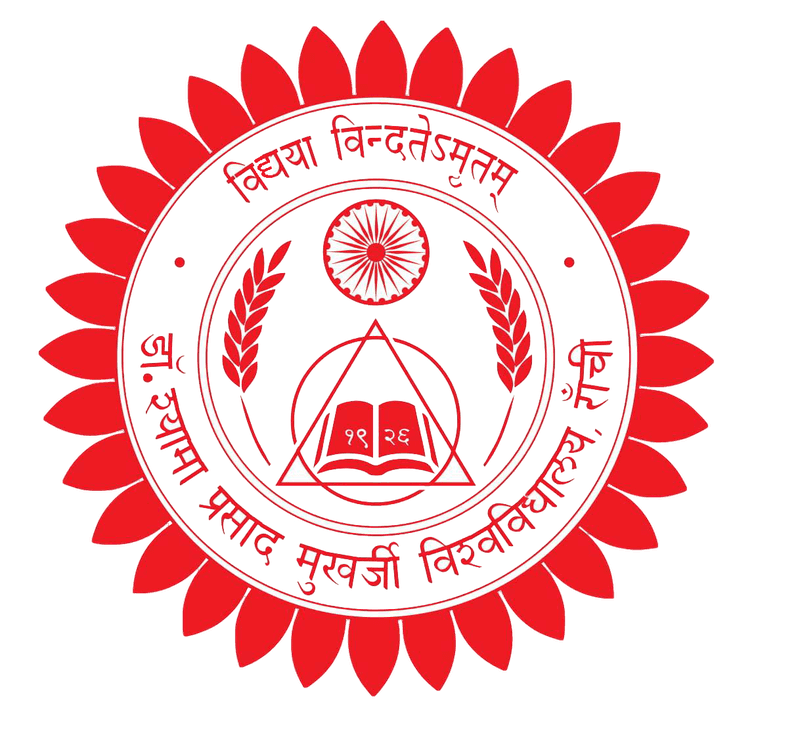


**MARKSHEETS**







****

**DECLARATION**

I hereby declare that the project entitled **“Decision Review System”** is an outcome of my efforts under the guidance of **Anchal Kumari.** The project is submitted to the Department of Information Technology, Dr. Shyama Prasad Mukherjee University Ranchi for the partial fulfilment of the Bachelor of Information Technology examination 2020-2023.

I further declare that the information and data given in the report are authentic to the best of my knowledge, this project report has not been submitted to any other University or institution for the award of any degree.

Date: Name: **Vishal Kumar**

Exam Roll no: **20B521603**

Reg. no: **DSPMU2020002095**

Session: **2020-2023**

# ACKNOWLEDGMENT

This thesis work has been an intellectually invigorating experience for me. I am sure that the knowledge and experience gathered during the course of this work will make me stand in good stead in future.

With immense pleasure and due respect, I express my sincere gratitude to In charge, DSPMU Ranchi, for all his support and co-operation in successfully completing this thesis work by providing excellent facilities.

I would also like to extend my sincere gratitude to all faculties’ members and staff for helping me in my college during my B.sc IT course.

I would like to take this opportunity to extend my sincere gratitude and thanks to my Pioneer **Anchal Kumari,** firstly for coming up with such an innovative thesis idea. He has not only made us to work but guided us to orient toward research. It has been real pleasure working under his guidance and it is chiefly his encouragement and motivation that has made this thesis a reality.

Last, but not the last I am heartily thankful to almighty God for showering his blessing forever during my entire life and also to my family members for providing me a great support.

# Table of Contents

**Name of the topics Pages No.**

1. Introduction 1

2. Objective of the project 1

3. Scope of the project 2

4. Project category 3

(a) Software Requirements

(b) Hardware Minimum Requirements

(c) Language Used

5. Literature Survey 4-5

(a) Umpire decision review system in cricket

(b)Decision making using image processing

(c) DRS in cricket

6. Existing System 5-6

7. Proposed System 7-8

(a) GUI for DRS

(b) Frame substraction

(c) Python image library

(d) Imutils module

8. Data Flow Diagram 9-13

(a) Developoment of GUI

9. Working Principal 13-14

10. Results 15-20

11. Coding 20-27

12. Testing 28-30

9.1. Introduction

9.2. Objectives

9.3. Methods

9.4. Steps

11. Conclusion 31

14. Future Scope 31

15. Bibliography 32

1. **Introduction**

This project is aimed at developing a “***Decision Review System***”.

In the game of cricket umpires have the authority to check whether the delivery is legal or not. They have the complete authority. Decision cannot be given by normal human eye there for the technology of decision renew system takes place. Umpires cannot have eye on each and every place on the field. Therefore it becomes necessary to use decision review system. As umpire cannot be right every time so we use decision review system for a fair game. So umpire simply ask for decision review system by giving signal of the square sign. Then the third umpire will look into this process. The decision review system must use on large scale for a fair play. The decision review system can also be opted by player. The important factor is that are considered here are television replays. Technology place a key role in every international sports in modern era. The decision gives the fair and proper result by using technology. And thus there are no controversies happening. By using the DRS people are knowing the rules of games. And this is how DRS place an important role in the system.

Thissystem involves all the features of the Decision Review System.

1. **Objective of the project:**

Our aim is to create a Decision review system (DRS) with theobjective of eliminating the howler – the poor decision takendue to nuances missed with the naked eye, which with assistance of this technology can be avoided. Our project willprovide aid with decisions like run-out, stumpout,net-cross,declaring a winner etc. Tkinter will be used to develop the Graphical User Interface (GUI) of DRS. To detect the decision or occurrence of event of the sport from the video we will be optimizing and using various functions such as frame subtraction using an inbuilt python module called OpenCV library.

1. **Scope of the project:**

The Decision Review System (DRS) is an ambitious and comprehensive project that seeks to transform decision-making processes across various domains. The primary goal of DRS is to introduce an efficient, transparent, and fair mechanism for reviewing critical decisions that impact individuals and communities.

The scope of the project is broad, covering multiple fields, including sports officiating, legal proceedings, healthcare, finance, and more. In the realm of sports, DRS aims to minimize controversies arising from incorrect umpire or referee calls by integrating advanced technologies like ball-tracking, ultra-motion cameras, and predictive analytics. This ensures fair outcomes and enhances the overall integrity of the sporting events.

In legal settings, the DRS project focuses on assisting judges and juries in making well-informed decisions. By leveraging natural language processing and data mining techniques, the system can analyze vast volumes of legal precedents, statutes, and case laws, providing valuable insights and reducing the risk of biased judgments.

In healthcare, DRS aims to enhance diagnostic accuracy and treatment decisions. By utilizing machine learning and medical imaging analysis, the system can aid medical professionals in identifying patterns and anomalies, leading to improved patient outcomes.

Furthermore, the financial sector can benefit from DRS by incorporating intelligent algorithms to review investment decisions, detect fraudulent activities, and ensure regulatory compliance.

The project's scope also encompasses the development of user-friendly interfaces that enable decision-makers to access and interpret DRS insights easily. Additionally, data privacy and security measures will be a crucial aspect to protect sensitive information and ensure trust in the system.

Overall, the Decision Review System project endeavors to usher in a new era of accurate, transparent, and data-driven decision-making across diverse domains, contributing to a more just and equitable society.

1. **Project category**: **REVIEW SYSTEM**

**4.1. Software Requirements:**

* OpenCV Library - 32- or a 64-bit computer, Windows 7,10or 11, macOS X 10.11 or higher, or Linux RHEL 6/7, Python2.7, 3.4, 3.5 or 3.6.
* Tkinter (8.6. 11)
* Imutils - Open CV Library, Python 2.7 or 3.Visual Studio Code IDE (1.61.1)

**4.2. Hardware Minimum Requirements:**

* Processor: Pentium 4 or above
* PC - Modern Operating System: 32- or a 64-bit Windows 7, 10 or 11, Mac OS X 10.11 or higher, 64-bit Linux: RHEL 6/7, 64-bit (almost all libraries also work in Ubuntu) x86 64-bit CPU (Intel / AMD architecture) 4 GB RAM 5 GB free diskspace.

**4.3. Language used:**

**Front End:** PYTHON

**2. LITERATURE SURVEY**

**2.1 Umpire DecisionReview System in Cricket**

This paper aims to produce a very cost effective and affordable computer mechanism that supports and facilitates the cricket umpire, runs at a low budget, has lower technical (software and hardware) requirements, and can be used in cricket tournaments at local district level, this indeed will train the network as well as improve the efficiency of the game from a grass root level. The module uses tools like Python, Tkinter for GUI, Pillow, OpenCV and Imultis packages for Python. In this project specifically, live feed of the instance can be fed through thecamera as an input stream for the software and thus decisions can be carried out in an instance, for the live audience. Ultimately saving precious time. Without any fancy equipment and with the help of just a computer system and a camera this system can run efficiently, thus making it compatible for all scale tournaments. This project is only an infant and there is always room for various kinds of improvement and modifications.

**2.4 Decision making using image processing**

This paper proposes a new approach for detecting the run out using stump detection and from the intersection of the bat and the crease line, named Artificial Intelligent third Umpire (AITU). The accuracy of decision making is improvedthereby reducing the third umpire work. If the on-field umpire within a fraction of a second was not able to make a decision, then the decision from the third umpire is sought. Then the third umpire views various TV replays at various angles and will declare the result as ‘out’or‘not out’. If the umpires are not sure about the decision that a batsman is ‘out’ or not, as they are deficient with more convincing evidence, the standard practice is to set free thebatsman, known as "the benefit of the doubt". However, thismay possess certain problems includingdelayinmatchwhichmay prove to be critical. The input to the proposed module is a video. A digital videoencompasses a succession of digital images displayed in quickprogression. The match video is processed to get the frames which are then used for making the decision. The foreground is extracted by performing background subtraction. Initially pre-processing is done to enhance the image quality. Frame segmentation is process of changing the representationoftheimage to make it easier for analysis. It is done on video frames to locate object boundaries like lines, curves, etc.

While the following system proves to be an accurate model to provide decision for run out without human intervention, unfortunately the same cannot be said for decision such as LBW detection and Catch detection.

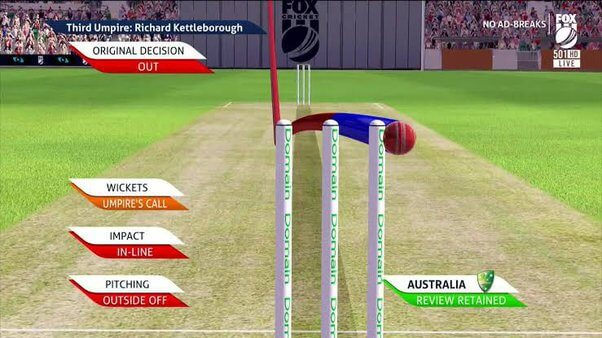
**2.6 Decision Review System in Cricket**

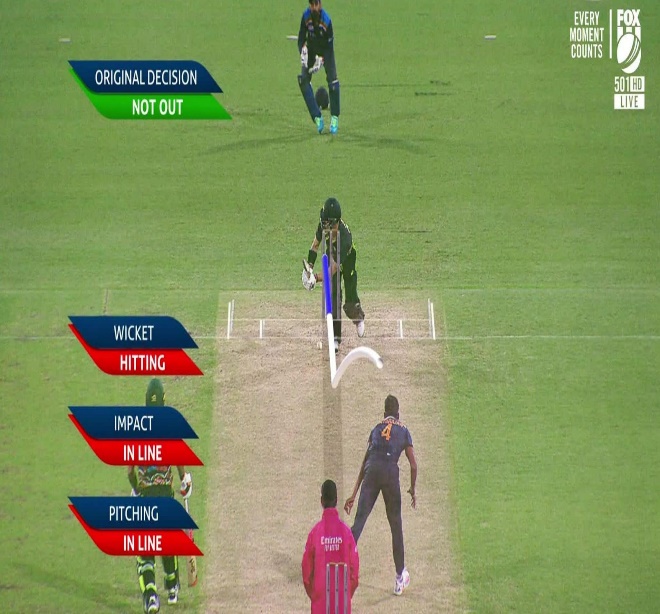
This paper suggests that the most common instances of referrals are for caught behind and Leg Before Wicket (LBW).As the final decision rides on chance of agreement of the field umpire along with, the DRS algorithms since this technologypredicts the nature of the trajectory of the delivery and such factors are dealt in very small measurements (in millimeters). Various errors included may change the final result for the same delivery if replayed again. The paper put forth themainidea that is to have a continuous real time feed clicked from the six cameras strategically placed in the stadium each operating at the specific frame rate which is enough to capture the pitched delivery to provide data points for mapping the virtual trajectory of the ball. Though, there is always a question of the accuracy with which the technologyworks as its decision cannot be referred again and is considered final. Hence, the accuracy of the technology used is limited to a certain range, but along with the human intervention in the decision making, this review system is bound to be the perfect solution to such a problem.

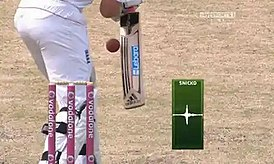
**3. EXISTING SYSTEM**

**3.1 Decision Review system in Cricket**

The decision review system in cricket comprises of various components such as Hotspot, Hawk-Eye, Snickometer, Ultra- Edge, etc. For the Hawkeye, the main goal is to have a continuous real time feed clicked from the six cameras strategically placed in the stadium each operating at the specific frame rate which is enough to capture the pitched delivery to provide data points for mapping the virtual trajectory of the ball. The third umpire refers to this technology when a referral for a LBW is sent upstairs to verify whether the ball was going on to hit the stumps if notinterrupted by the batsmen leg.

****

****

****

1. **PROPOSED SYSTEM**

Traditionally, in various sports the final judgement is givenby the referee or the umpire. While usually most of these decisions are correct, often there are discrepancies behind these decisions since human judgement is involved which has the potential to change the final result of the game. Our System is basically an aid to the umpire in case of such discrepancies where in only a good quality camera and a system to operate the software is needed. What this human aid does is, it operates on frame rate to give a clear picture ofthe exact moment where the decision needs to take place and the final call is given by the referee themselves.

**For the module:**

Our project will be implemented using Python and Tkinter, OpenCV library, Python Imaging Library, Imutils, Pillow module etc. Decisions like stump out, run out etc. can be decided by altering the source video’s frame rate for the which, we will be optimizing and using frame subtraction using OpenCV library.

**4.1 GUI of DRS:**

Tkinter is used to make GUI of the Decision Review Systemand various computer vison algorithms that give decisions based on different criteria. Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUImethods, Tkinter is the most commonly used method. Being a Python binding to the Tk GUI toolkit, it gives standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter also includes standard GNU/Linux, Microsoft Windows and mac OS installs of Python.

**4.2 Frame Subtraction:**

Frame subtraction will be done using OpenCV (OpenSource Computer Vision Library) which is a library of programming functions mainly aimed at real-time computer vision. OpenCV is an open-source library which is very useful for computer vision applications such as video analysis, CCTV footage analysis and image analysis. OpenCV supports a widevariety of programming languages like Python, C++, Java.It can process images and videos to identify objects, faces, oreven the handwriting of a human. OpenCV is written by C++ and has more than 2,500 optimized algorithms. When we create applications for computer vision that we don’t want to build from scratch we can use this library to start focusing on real world problems. There are many companies using this library today such as Google, Amazon, Microsoft and Toyota. Many researchers and developers contribute.Wecaneasily install itin any OS like Windows, Ubuntu and Mac OS. With reference to our project, OpenCV library will be majorly used to bring changes to the frame i.e., increasing or decreasing the frame speed.

**4.3 Python Imaging Library:**

Python Imaging Library (expansion of PIL) for which Pillow module is to be installed, is the de facto image processing package for Python language. It incorporates lightweight image processing tools that aids in editing, creating and saving images and with respect to our project, it will load images from an array.

**4.4 Imutils Module:**

Imutils which is basically a series of convenience functionsto make basic image processing functions such as translation, rotation, resizing, skeletonization, and displaying images easier with OpenCV will also be used for the implementation.

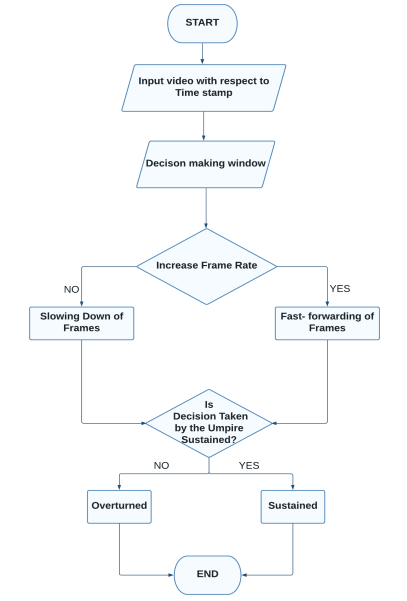
1. **Data Flow Diagram**

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an Information System. A data flow diagram can also be used for the visualization of Data Processing. It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context-level DFD is then "exploded" to show more detail of the system being modeled.

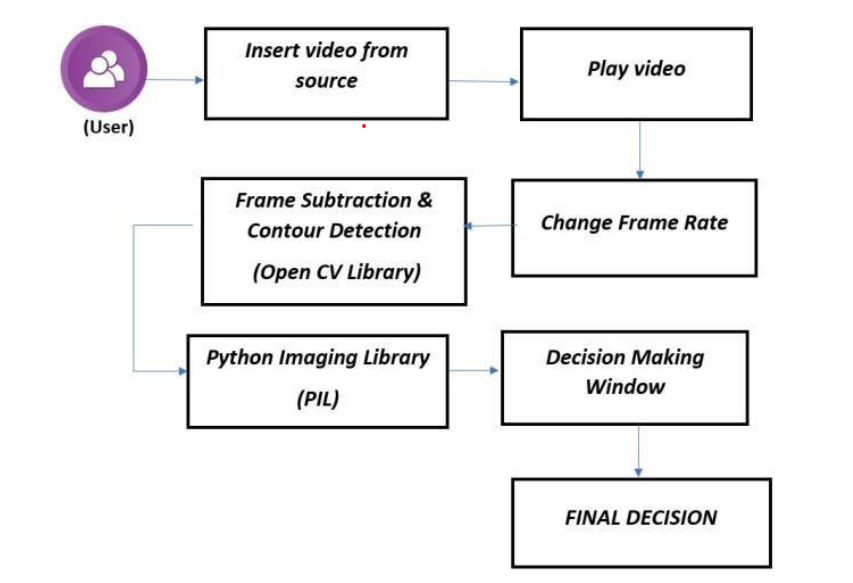
A DFD represents flow of data through a system. Data flow diagrams are commonly used during problem analysis. It views a system as a function that transforms the input into desired output. A DFD shows movement of data through the different transformations or processes in the system.

Dataflow diagrams can be used to provide the end user with a physical idea of where the data they input ultimately has an effect upon the structure of the whole system from order to dispatch to restock how any system is developed can be determined through a dataflow diagram. The appropriate register saved in database and maintained by appropriate authorities.

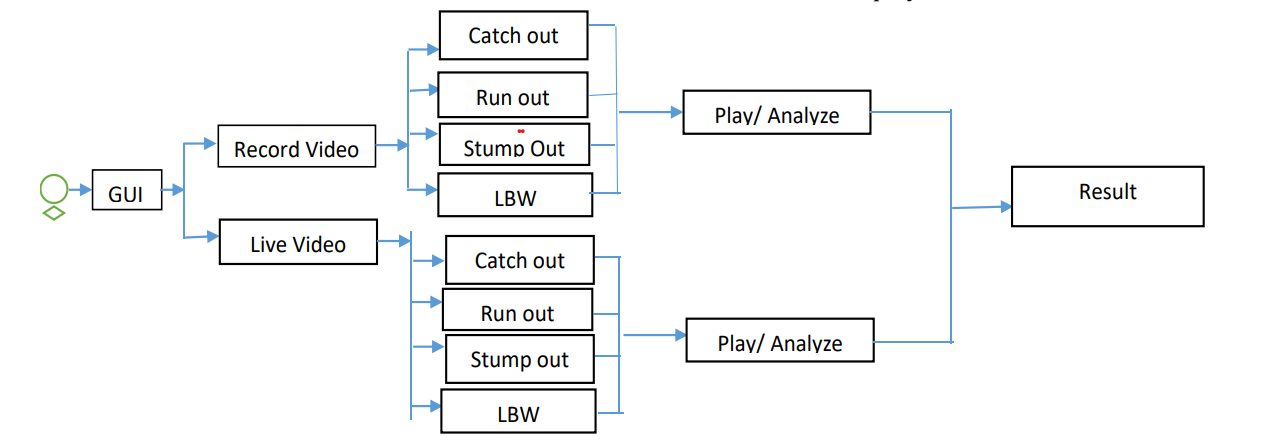
* 1. **Data Flow Diagram of the Decision Review System (DRS)**

****

In the system, the video will be added according to the time frame where in the decision making is required. Then the sport is to be selected accordingly following which a decisionwindow pop-up will appear where there will be options to slow or make the video go faster by altering the frame rate and thus it will be decided if the decision made by the umpire is to be sustained or overturned. System Architecture

****

The project has been implemented using tools like Tkinter for GUI, Pillow, and OpenCV. Tkinter is a GUI framework thatis built into the Python standard library that has been usedfor the front end. For the backend the python programminglanguage has been used. At first the user will be selecting a sport for which the decision has to be made. Then a video will be inserted from the source destination. Once inserted we can play the video with different frame rates available. This could be done using various modules and tools as mentioned. OpenCV is a hugeopen-source library for various sectors which has been primarily used for image processing. It will be used to process images and videos. Further Tkinter also relies on Pillow for working with images. Pillow is basically a fork of the Python Imaging Library and can be imported in a Python console as PIL. It incorporates lightweight image processing tools that aids in editing, creating and saving images. Hence by slowing downthe frame rate a collective decision (Human Aided) could be made. A decision-making window will appear which would have several options depending on the sport which had been chosen earlier and the final decision will be displayed.

****

Development of GUI

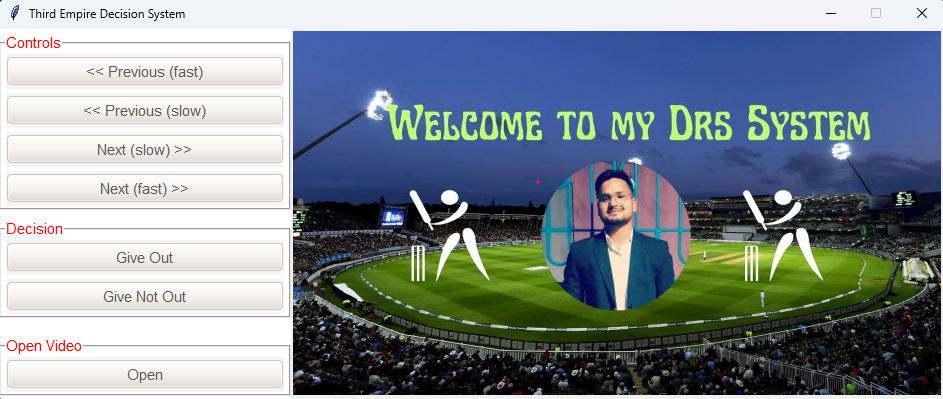
For the development of Graphical User Interface (GUI) we used the Tkinter library of python in which around 18 windows were created catered to specific functions in whichthe breakdown is as follows:

Cricket:

Cricket → Upload Video → main decision window frames are altered → Choose Decision → GIF pop-up window → Displayof decision.

**Working Principal:**

By using our decision review system project we can easily find whether the batsman is out or not out. Not only that, we can even check other decisions. In this system we first take input in the form of video. Then we process the video and then we analyse the video and therefore we are ready to make decision. After taken the input we process the video by frame by frame and thus it becomes easy to make result. And thus after making result we display it on output screen.

****

Screen

Buttons

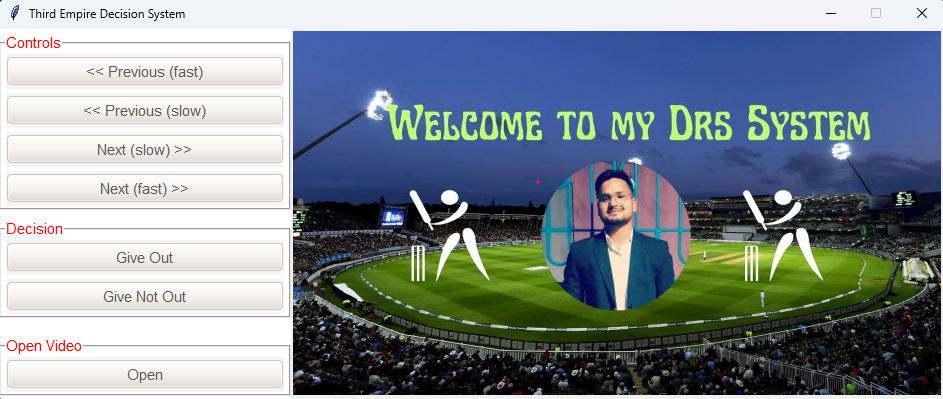
Here screen helps to display or stream the video which is uploaded as a source video .It helps to see and determine every aspect of DRS like out , not out , six , four and so on .

Here buttons helps to control the streaming video and to give their decision according to streaming video observation.

**Results:**

**Drs GUI:**

It Provides facilities to give any decision like Four , Six , Out and Not out. We have used Tkinter to make GUI of the Decision review system and various computer vision algorithms are used to give these decisions based on criteria’s



In our decision review system project we use a source. the source video will be analysed and thus some certain checks will be performed on it by checking each frame of the video. For making a correct decision whether the batsman is out or not out we need to analyse the video that has been recorded from the live camera.

****

Catch Out Condition

The catch out decision is determined by playing the video in slow motion also in fast motion based on further conditions. If the player touches the ball to ground then batsman is not out. But if the player takes safe catch without touch to ground then the batsman will not be given out. The striker is out Caught if a ball delivered by the bowler, not being a No ball, touches his/her bat without having previously been in contact with any fielder, and is subsequently held by a fielder as a fair catch, as before it touches the ground.

Run Out Condition

To determine the run-out decision the system plays the selected video in slow motion as well as in fast motion. If the wicketkeeper Run out: To determine the run-out decision the system plays the selected video in slow motion as well as in fast motion. . A run out usually occurs when the batsmen are attempting to run between the wickets, and the fielding team succeed in getting the ball to one wicket before a batsman has crossed the crease line near the wicket.

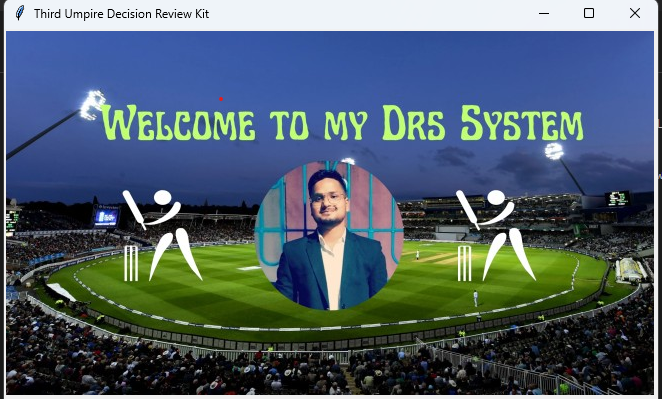
Stump Out Decision

The Stump-out decision is determined by playing the video in slow motion also in fast motion based on further conditions. Stumped is a method of dismissing a batsman in cricket, which involves the wicket-keeper putting down the wicket while the batsman is out of his ground. (The batsman leaves his ground when he has moved down the pitch beyond the popping crease, usually in an attempt to hit the ball). If player comes inside crease before keeper hits the stump then player is not out. If player fail to do so then he is out.

Leg before wicket is one of the ways in which a batsman can be dismissed in the sport of cricket. Following an appeal by the fielding side, the umpire may rule a batter out lbw if the ball would have struck the wicket but was instead intercepted by any part of the batter's body.

**Step by Step Working of the module**

* On running the module, we first come across the starting window where the user will se a welcome page and some clickable buttons , which will help the user to take decisions and to see the recored video with slow and fast motion according to need.
* After when we click on preview or next buttons then the video will play according to our need with the help of clickable buttons.
* After that decision reviewer or observer observe every aspect that happened in game, and come to on some conclusion.
* Then the third empire gives his decision with the clickable button means, out ,not out, six, four etc.
* After giving the decision the window will show a dicision pending popup silde , then it shows sponsored popup slide and then show the actual decided decision.

****

**Welcome screen**

****

**Button screen**



**Decision pending**



**Sponsered popup slide**









**Coding**

import tkinter

from tkinter import ttk,filedialog

import cv2

import PIL.Image, PIL.ImageTk

from functools import partial

import threading

import imutils

import time

from ttkthemes import themed\_tk as tk

stream = None

file\_path = ''

def open\_file():

global stream

global file\_path

file\_path = filedialog.askopenfilename()

stream = cv2.VideoCapture(file\_path)

def play(speed):

global flag

print(f"You clicked on play. Speed is{speed}")

# play the video in reverse

frame1 = stream.get(cv2.CAP\_PROP\_POS\_FRAMES)

stream.set(cv2.CAP\_PROP\_POS\_FRAMES,frame1 + speed)

grabbed,frame = stream.read()

if not grabbed:

exit()

frame = imutils.resize(frame,width=set\_width,height=set\_height)

frame = PIL.ImageTk.PhotoImage(image= PIL.Image.fromarray(frame))

canvas.image = frame

canvas.create\_image(0,0,image=frame, anchor=tkinter.NW)

canvas.create\_text(134,26,fill="yellow",font="Times 26 bold",text="Decision Pending")

def pending(decision):

# display decision pending image

frame = cv2.cvtColor(cv2.imread("pending.png"),cv2.COLOR\_BGR2RGB)

frame = imutils.resize(frame,width=set\_width,height=set\_height)

frame = PIL.ImageTk.PhotoImage(image=PIL.Image.fromarray(frame))

canvas.image = frame

canvas.create\_image(0,0,image=frame,anchor=tkinter.NW)

# display wait for 1.5 second

time.sleep(1.5)

# display sponser image

frame = cv2.cvtColor(cv2.imread("sponsor.png"),cv2.COLOR\_BGR2RGB)

frame = imutils.resize(frame,width=set\_width,height=set\_height)

frame = PIL.ImageTk.PhotoImage(image=PIL.Image.fromarray(frame))

canvas.image = frame

canvas.create\_image(0,0,image=frame,anchor=tkinter.NW)

# display wait for 2.0 second

time.sleep(2.0)

# display out/not out image

if decision == 'Out':

decisionImg = "out.png"

else:

decisionImg = "not\_out.png"

frame = cv2.cvtColor(cv2.imread(decisionImg),cv2.COLOR\_BGR2RGB)

frame = imutils.resize(frame,width=set\_width,height=set\_height)

frame = PIL.ImageTk.PhotoImage(image=PIL.Image.fromarray(frame))

canvas.image = frame

canvas.create\_image(0,0,image=frame,anchor=tkinter.NW)

def out():

thread = threading.Thread(target=pending,args=("Out",))

thread.daemon = 1

thread.start()

print("Player is Out")

def not\_out():

thread = threading.Thread(target=pending,args=("Not Out",))

thread.daemon = 1

thread.start()

print("Player is Not Out")

# width and height of a main screen

set\_width = 650

set\_height = 368

# Gui

# Window

# window theme

window = tk.ThemedTk()

window.get\_themes()

window.set\_theme("radiance")

# window configure

window.configure(bg='white')

window.resizable(width=False,height=False)

window.title("Third Empire Decision System")

# canvas

cv\_img = cv2.cvtColor(cv2.imread("welcome.jpg"),cv2.COLOR\_BGR2RGB)

canvas = tkinter.Canvas(window,width=set\_width,height=set\_height)

photo = PIL.ImageTk.PhotoImage(image=PIL.Image.fromarray(cv\_img))

image\_on\_canvas = canvas.create\_image(0,0,anchor=tkinter.NW,image=photo)

canvas.grid(row = 0,column=1,rowspan=4)

# Buttons to control playback

# frame1

frame1 = tkinter.LabelFrame(window,text='Controls',bg='white',fg='red')

btn = ttk.Button(frame1,text="<< Previous (fast)",width=30,command=partial(play, -25))

btn.grid(row=0,column=0,padx=5,pady=2)

btn = ttk.Button(frame1,text="<< Previous (slow)",width=30,command=partial(play, -2))

btn.grid(row=1,column=0,padx=5,pady=2)

btn = ttk.Button(frame1,text="Next (slow) >>",width=30,command=partial(play, 2))

btn.grid(row=2,column=0,padx=5,pady=2)

btn = ttk.Button(frame1,text="Next (fast) >>",width=30,command=partial(play, 25))

btn.grid(row=3,column=0,padx=5,pady=2)

frame1.grid(row=0,column=0)

# end of frame 1

# Buttons to decison

# frame2

frame2 = tkinter.LabelFrame(window,text="Decision",bg='white',fg='red')

btn = ttk.Button(frame2,text="Give Out",width=30,command=out)

btn.grid(row=1,column=2,padx=5,pady=2)

btn = ttk.Button(frame2,text="Give Not Out",width=30,command=not\_out)

btn.grid(row=2,column=2,padx=5,pady=2)

frame2.grid(row=1,column=0)

# end of frame 2

# Button to open file

# frame3

frame3 = tkinter.LabelFrame(window,text="Open Video",bg='white',fg='red')

btn = ttk.Button(frame3,text="Open",width=30,command=open\_file)

btn.grid(row=0,column=0,padx=5,pady=2)

frame3.grid(row=3,column=0)

window.mainloop()

**Testing**

* 1. **INTRODUCTION**

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements.

The main purpose of testing is to detect errors and error-prone areas in a system. Testing must be thorough and well-planned. A partially tested system is as bad as an untested system. And the price of an untested and under-tested system is high.

The implementation is the final and important phase. It involves user-training, system testing in order to ensure successful running of the proposed system. The user tests the system and changes are made according to their needs. The testing involves the testing of the developed system using various kinds of data. While testing, errors are noted and correctness is the mode.

**OBJECTIVES OF TESTING:**

The objectives of testing are:

* Testing is a process of executing a program with the intent of finding errors.
* A Successful test case is one that uncovers an as- yet-undiscovered error.

System testing is a stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as per the user need, before the live operation commences. As stated before, testing is vital to the success of a system. System testing makes a logical assumption that if all parts of the as system are correct, the goal will be successfully achieved. A series of tests are performed before the system is ready for the user acceptance test.

**TESTING METHODS**

System testing is the stage of implementation. This is to check whether the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. The candidate system is subject to a variety of tests: on line response, volume, stress, recovery, security and usability tests. A series of tests are performed for the proposed system is ready for user acceptance testing.

**The Testing Steps are:**

* **Unit Testing**

Unit testing focuses efforts on the smallest unit of software design. This is known as module testing. The modules are tested separately. The test is carried out during programming stage itself. In this step, each module is found to be working satisfactory as regards to the expected output from the module.

* **Integration Testing**

Data can be lost across an interface. One module can have an adverse effect on another, sub functions, when combined, may not be linked in desired manner in major functions. Integration testing is a systematic approach for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface. The objective is to take unit tested modules and builds program structure. All the modules are combined and tested as a whole.

* **Validation**

At the culmination of the integration testing, Software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test begin in validation testing. Validation testing can be defined in many ways, but a simple definition is that the validation succeeds when the software functions in a manner that is expected by the customer. After validation test has been conducted, one of the three possible conditions exists.

1. The function or performance characteristics confirm to specification and are accepted.
2. A deviation from specification is uncovered and a deficiency list is created.
3. Proposed system under consideration has been tested by using validation test and found to be working satisfactory.

* **Output Testing**

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format. The output format on the screen is found to be correct. The format was designed in the system design time according to the user needs. For the hard copy also; the output comes as per the specified requirements by the user. Hence output testing did not result in any correction for the system.

* **User Acceptance Testing**

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes whenever required.

This is done in regard to the following point:

1. Input Screen Design
2. Output Screen Design
3. Format of reports and other outputs.
4. **Conclusion**

The main objective of our project is to assist the umpire in game of cricket to make fair decision in efficient way. Our project discusses the use of computer vision algorithms to extract frames and to detect, tracked the ball from the provided source video. Python helps to visualize the results of various decisions which makes the UI more attractive and interactive. Also use of single smartphone camera makes the system cost efficient and easy to use. Counting semantic structure, cinematography, framing, iconography, focus management and feedback, there’s quite a lot it takes when creating Avery efficient third umpire review system so everything we’ve learned here will help us create a simple and very efficient DRS system applicable in big range of outdoor sports. There is always that instinct of patriotism in sports with fair decisions at the lesser level of sports tournaments to be played and this DRS can provide them a very cost effective and quality decision review system that will boost up the game spirit of each player. Without any fancy equipment and with the help of a computer system and a camera this system can run efficiently, thus making it compatible for all scale tournaments. This project is only an infant and there is always room of improvement and modifications making it more ready to go out and rock the gully cricket world. DRS technologies have been associated generally with very costly applications. Certain decision review systems are not so cost effective and highly unreliable as they consume too much data and are time consuming.

**FUTURE SCOPE**

At present, this system works when the exe. File along with python are installed in the device. Further a feature of live camera feed along with a website can be implemented for the module to work on devices which doesn’t necessary have python installed on them.

1. **BIBLIOGRAPHY**

* https://www.python.org
* www.wikipedia.com
* https://www.tutorialpoint
* **youtube.com**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*