

**A CASE STUDY (IEEE Format)**

---

**Software Requirements Specification Document**

**Version1.0**

***Mouse and Webcam Controller***

**Submitted By**

Chetanveer Singh 102117053

Harneet Kaur 102117052

Ira Gupta 102117057

Vansh Sandhir 102117060

# TABLE OF CONTENTS

1.	<a href="#"><u>Introduction</u></a>	3
1.1	<a href="#"><u>Purpose of this Document</u></a>	3
1.2	<a href="#"><u>Scope of the Development Project</u></a>	3
1.3	<a href="#"><u>Definitions, Abbreviations and Acronyms</u></a>	4
1.4	<a href="#"><u>References</u></a>	5
1.5	<a href="#"><u>Overview</u></a>	7
2	<a href="#"><u>Overall Description</u></a>	8
2.1	<a href="#"><u>Product Perspective</u></a>	9
2.2	<a href="#"><u>Product Functions</u></a>	9
2.3	<a href="#"><u>User Characteristics</u></a>	10
2.4	<a href="#"><u>General Constraints, Assumptions and Dependencies</u></a>	11
3	<a href="#"><u>Specific Requirements</u></a>	12
3.1	<a href="#"><u>Purpose, Scope and Overview</u></a>	12
3.2	<a href="#"><u>Functional Requirements</u></a>	12
3.3	<a href="#"><u>User Interface Requirements</u></a>	12
3.4	<a href="#"><u>Performance Requirements</u></a>	13
3.5	<a href="#"><u>Hardware Requirements</u></a>	13
3.6	<a href="#"><u>Software Requirements</u></a>	14
3.7	<a href="#"><u>General Constraints</u></a>	14
3.8	<a href="#"><u>Other Non Functional Requirements</u></a>	14
3.9	<a href="#"><u>Quality Attributes</u></a>	15
4	<a href="#"><u>Change History</u></a>	15
5	<a href="#"><u>Document Approvers</u></a>	15
6	<a href="#"><u>Use Case Diagram</u></a>	16
7	<a href="#"><u>Use Case Template</u></a>	17
8	<a href="#"><u>Activity Diagram</u></a>	18

## 1.Introduction

### 1.1Purpose of this Document:

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client ,team and audience see the product and its functionality.

### 1.2 Scope of the Development Project:

The goal is to design a Human-Computer Interaction(HCI) based multi-utility program including Mouse and Webcam controller. The main objective is to find the solution for the finger-tracking in the real world and control the cursor of a computer without touching the mouse physically. In this system, the user would make use of the webcam and some algorithms to control the cursor operations. This work presents the implementation and the analysis of real time tracking of fingers which are involved in making a gesture, so that gestures can be used in various applications of mouse like movement, single-click, double-click, right-click, scrolling, face-detection and screenshot. Color identification and tracking would be simpler than using algorithms like neural networks.

The software must be able to perform the following operations:

**Movement/Scrolling :** It must be able to scroll the screen in the similar way as it could have been done by using the scroll wheel of the physical mouse. The cursor movements should be smooth enough.

**Left-click/Right-click:** It must be able to capture each frame for the termination of the different clicks performed by the physical mouse. This may include right-click, left-click, tapping, or double-click with the specific tip-Ids assigned to them.

**Screenshot:** It must be able to take the screenshot if both the index finger and the middle finger are up with the thumb at right angle and the distance between them is less than 40px, our system would be able to take the perfect screenshot.

**Color Detection:** It must be able to read the image, define the boundaries and would be able to detect colour using cv2.inRange(). Bit-wise and operation would be performed with the input image and the colours would be red as BGR color-space.

**Face Detection:**It must be able to detect face by using OpenCV containing many pre-trained classifiers for face, eyes, smiles, etc. The detection works only on grayscale images. So it would be important to convert the colour image into grayscale. We would use the face coordinates to detect the image using rectangles.

Initially we plan to implement these functionalities withan intended audience of 30 people, out of which 2 would be the staff members and remaining would be our fellow class students. This would be a part of the **Pilot Phase**. Once the Pilot Phase is successful thenwe plan to implement it in other labs across the institute and eventually we plantoextendtheHCIBasedMulti-UtilitySystemincludingMouse and WebcamControllerto a wide variety of applications.

The scope of this system is not just limited to the university campus only as the same mechanism can be reused in other campuses, households, companies, offices as well. This system can also be implemented in the technical sector where virtual mouse can take place of the physically available mouse. When a person sitting at his PC, trying to use his mouse attached to the system find it difficult, this scenario could be easily overcome by using our proposed AI virtual mouse system. It would be prevalent in the coming age, where most of the companies would try to shift and implement smart and virtual access control mechanisms.

### **1.3 Definitions, Abbreviations and Constraints:**

#### **Definitions**

Table1 gives explanation of the most commonly used terms in this SRS document.

**Table1: Definitions for most commonly used terms**

S.No.	Term	Definition
1	GUI	It is a friendly visual environment that allows the user to perform any action without having to have programming knowledge. An example of the GUI are the Windows, MacOS or Android environments, thanks to which commands can be sent through gestures or mouse movements, without the need to enter any code.
2	HIC	Human-computer interaction is a multidisciplinary study that focuses on the interaction between people and computers as well as the design of the computer interface. Factors to take into account include the user capabilities and cognitive processes, personality, experience, motivation, and emotions.
3	HSV	The HSV color space (hue, saturation, value) is often used by people who are selecting colors (e.g., of paints or inks) from a color wheel or palette, because it corresponds better to how people experience color than the RGB color space does.
4	IDE	An integrated development environment (IDE) is a software application that helps programmers develop software code efficiently. It increases developer productivity by combining capabilities such as software editing, building, testing, and packaging in an easy-to-use application.
5	MHI	The motion history image (MHI) is a static image template helps in understanding the motion location and path as it progresses.[1] In MHI, the temporal motion information is collapsed into a single image template where intensity is a function of recency of motion

#### **Abbreviations**

Table2 gives the full form of most commonly used mnemonics in this SRS document.

**Table2:Full form for most commonly used mnemonics**

S.No.	Mnemonic	FullForm
1	GUI	Graphical User interface
2	LED	Light Emitting Diodes
3	HCI	Human Computer Interaction
4	MHI	Motion History Images
5	IDE	Integrated Development Environment
6	OpenCV	Open Source Computer Vision
7	HSV	Hue,Saturation and Values
8	RGB	Red,Green and Blue

#### **1.4 References:**

- [1] Zhang, Zhengyou, et al. "Visual panel: virtual mouse, keyboard and 3D controller with an ordinary piece of paper." *Proceedings of the 2001 workshop on Perceptive user interfaces*. 2001.
- [2] Zhou, Qian, George Fitzmaurice, and Fraser Anderson. "In-depth mouse: Integrating desktop mouse into virtual reality." *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 2022.
- [3] Reddy, Vantukala VishnuTeja, et al. "Virtual Mouse Control Using Colored Finger Tips and Hand Gesture Recognition." *2020 IEEE-HYDCON*. IEEE, 2020.
- [4] Robertson, Paul, Robert Laddaga, and Max Van Kleek. "Virtual mouse vision based interface." *Proceedings of the 9th international conference on Intelligent user interfaces*. 2004.
- [5] Shibly, Kabid Hassan, et al. "Design and development of hand gesture based virtual mouse." *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*. IEEE, 2019.

[6] Pham, Duc-Minh, and Wolfgang Stuerzlinger. "Is the pen mightier than the controller? a comparison of input devices for selection in virtual and augmented reality." *Proceedings of the 25th ACM Symposium on Virtual Reality Software and Technology*. 2019.

[7] Pham, Duc-Minh, and Wolfgang Stuerzlinger. "Is the pen mightier than the controller? a comparison of input devices for selection in virtual and augmented reality." *Proceedings of the 25th ACM Symposium on Virtual Reality Software and Technology*. 2019.

[8] Shetty M, Daniel CA, Bhatkar MK, Lopes OP. Virtual mouse using object tracking. In 2020 5th International Conference on Communication and Electronics Systems (ICCES) 2020 Jun 10 (pp. 548-553). IEEE.

[9] Patil, Nilesh, Gaurav Sali, and Nikhil Lokhande. "Mouse on Finger Tips using ML and AI." (2019).

[10] Mali, S. P., et al. "Finger Mouse Movement." *International Journal of Engineering and Management Research (IJEMR)* 8.2 (2018): 5-7.

### **1.5 Overview:**

The remaining sections of this document provide a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product. General description of the project is discussed in section 2 of this document. Section 2 gives the functional requirements, data requirements and constraints and assumptions made while designing the multi-utility system. It also gives the user viewpoint of product use. Section 3 gives the specific requirements of the product. Section 3.0 also discusses the external interface requirements and gives detailed description of functional requirements.

## 2. Overall Description

### 2.1 Product Perspective:

.In the overview of the system, the suggested environmental setup of the system is described as well as the architecture of the system. The suggested environmental setup for the system is shown in the figure below.



The user's hand should be at least 24cm from the webcam and the room should be well lit. The webcam must not be moving and if it is shifted the program should be restarted. In the figure above, the user is only using one hand to operate the program and the other hand must not be in front of the webcam. This application uses color information to detect skin color from the webcam image. The environment in which this application is being used must be well lit. The user's hand should be free from any paint or markings that would obscure skin detection. The light source should be white and not be colored, as this would cause the appearance of skin color to change. There should not be any moving persons in the background the application is being used. The lighting condition of the room should be constant. If the lighting condition changes, the program must be restarted.

## **2.2 Product Functions:**

The product should be able to perform the following operations:

- It must be able to authenticate the card user by matching the ID no. / PSRN no. and the access code against the values stored in the database.
- It must be able to check the lab/room status by querying the database for any reservation requests made earlier.
- It must be able to record the user's presence by writing the user's ID no. / PSRN no. in the corresponding database table. Thus for one swipe, two write operations will be performed: one into the central repository and other into the backup database server. For each swipe the time in/out will also be recorded and the total time spent in the lab will be computed by subtracting the time when user entered the lab with the time when the user came out of the lab.
- The software must be able to update the access privileges onto a particular user's card and the database where the privileges themselves will be modifiable only by the system administrators (or some authorized staff members).
- The software must be able to determine whether a particular user has been denied access from a particular lab due to some policy violation. The results of this operation will be viewable by the security officer only.

## **2.3 User Characteristics:**

The two main categories of users for this product are typical users and expert users. The typical user would be anyone who uses the application for their personal uses such as navigating the computer cursor. This product was originally intended of this type of user and the product would be most frequently used by them. The expert user would incorporate the application into their design or product. Robotic engineers would be classified as an expert user, since they would incorporate this application into their robot control application. Computer game developers would also be considered as an expert user since they can also incorporate this application into their game design.

The goal is to design software for a HCI based Multi-Utility System including mouse and Web Cam controller for different users.

As one can see from the list, each user will have different educational background and expertise level in using the system. Our goal is to develop software that should be easy to use for all types of users. Thus while designing the software one can assume that each user type has the following characteristics:

The user is computer-literate and has little or no difficulty in using webcam to access information such as hand gestures of human.

In order to use webcam it is not required that a user be aware of the internal working of computer. Also the availability of the monitor or display is important as to interact with our proposed generalised system. Furthermore, the proposed method can be developed to handle the keyboard functionalities along with the mouse functionalities, virtually which is another future scope of Human Computer Interaction (HCI).



## **2.4 General Constraints, Assumptions and Dependencies:**

The following list presents the constraints, assumptions, dependencies or guidelines that are imposed upon implementation of the HCI based Multi-Utility System including Mouse and Webcam Control:

### **a) Webcam & Variables Initialization**

On the early stage of the recognition phase, the program will initialize the required variables which will be used to hold different types of frames and values where each are will be used to carry out certain task. Furthermore, this is the part where the program collects the calibrated HSV values and settings where it will be used later during the transitions of Binary Threshold.

### **b) Real Time Image Acquisition**

The real time image is captured by using the webcam by using (cv::VideoCapture cap(0);), where every image captured are stored into a frame variable (cv::Mat), which will be flipped and compressed to a reasonable size to reduce process load.

### **c) Frame Noise Filtering**

Similar to the noise filtering during the calibration phase, Gaussian filters will be applied to reduce the existing noise of the captured frames. This can be done by using GaussianBlur(InputArray src, OutputArray dst, Size ksize, double sigmaX, double sigmaY=0, int borderType=BORDER\_DEFAULT ).

### **d) HSV Frame Transition**

The captured frame require to be converted from a BGR format to a HSV format. Which can be done by using cvtColor(src, dst, CV\_BGR2HSV).

### **e) Binary Threshold Transition**

The converted HSV frame will undergone a range check to check if the HSV values of the converted frame lies between the values of the HSV variables gathered during the calibration phase. The result of the range check will convert the frame into a Binary Threshold, where a part of the frame will set to 255 (1 bit) if the said frame lies within the specified HSV values, the frame will set to 0 (0 bit) if otherwise

### **f) Colour Combination Comparison**

After obtaining results from Morphological Transformation process, the program will calculate the remaining number of objects by highlighting it as blobs, this process requires cvBlob library, which is an add-on to OpenCV. The results of the calculation will then send for comparison to determine the mouse functions based on the colour combinations found within the captured frames

### **g) Execution of Mouse Action**

The program will executes the mouse actions based on the colours combinations exist in the processed frame. The mouse actions will perform according to the coordinates provided by the program, and the program will continue on acquire and process the next real-time image until the users exit from the program.

### **3. Specific Requirements**

#### **3.1 Purpose, Scope and Overview:**

- The purpose of the project is to make a virtual mouse which has no physical connection with the computer.
- It works with the help of a web camera. It has a lot advantages over the existing physical mouse.
- It is portable. It occupies less space and is very comfortable to the users.
- In this system the camera continuously takes images and the images are analyzed using various image processing techniques and mouse input is given to a particular screen position.
- The user must have a particular color in his hand so that when the web camera takes image it must be visible in the image. This color is detected from the image pixel and the pixel position is mapped into mouse input.
- Depending upon the size of the image taken by camera various scaling techniques are used because the pixel position in the image will not have a correspondence with screen resolution.

#### **3.2 Functional Requirements:**

The various Functional requirements of the system can be summarized as follows:

- The system is will provide a good user interface through which the user can interact with the system. The virtual mouse enables the user to give mouse inputs.
- To work with media packages Java Media Framework should be installed.
- A text editor like Notepad also needed for editing the application.

#### **3.3 User Interface Requirement**

User Education and Training : -

To achieve the objectives and benefits expected from the computer based system, it is essential for people who will be involved to be confident of their role in the new system. This involves them in understanding the overall system. As the system becomes more complex the need for education and training is more and more important. Education of the user should really have taken place much earlier in the project when they were being involved in the investigation and design work.

System testing is an expensive but critical process that can make as much as fifty percentage of the budget of the program development.

The common view of testing held by the user is that it is performed to prove that there are no errors in the program. Therefore, the most practical approach is with understanding that testing is the process of executing programs with the intention of finding errors.

### **3.4Performance Requirements:**

Considering the interactive nature of the task the system must have the following characters.

- Minimum response time
- Efficient CPU utilization
- Less Memory space
- High reliability
- High flexibility
- User friendly

### **3.5Hardware Requirement:**

The following describes the hardware needed in order to execute and develop the Virtual Mouse application:

- Computer Desktop or Laptop

The computer desktop or a laptop will be utilized to run the visual software in order to display what webcam had captured. A notebook which is a small, lightweight and inexpensive laptop computer is proposed to increase mobility.

System will be using:

Processor : Core2Duo

Main Memory : 4GB RAM

Hard Disk : 320GB

Display : 14" Monitor

- Webcam

Webcam is utilized for image processing, the webcam will continuously take image in order for the program to process the image and find pixel position.

### **3.6 Software Requirements:**

The following describes the software needed in order to develop the VirtualMouse application:

- C++ Language

The coding technique on developing the Virtual Mouse application will be the C++ with the aid of the integrated development environment (IDE) that are used for developing computer programs, known as

the Microsoft Visual Studio. A C++ library provides more than 35 operators, covering basic arithmetic, bit manipulation, indirection, comparisons, logical operations and others.

- Open CV Library

OpenCV are also included in the making of this program.

OpenCV (Open Source Computer Vision) is a library of programming functions for real time computer vision. OpenCV have the utility that can read image pixels value, it also have the ability to create real time eye tracking and blink detection.

Software will be using:

OS : Window 7 Ultimate 64-bit

Language : C++

Tool Used : Open CV and CMake

### **3.7General Constraints:**

As the clients is not used to an automated environment they didn'timpose any stringent constraints over the system. But they put they forth some important comment that is desirable for the proposed system.

- They want an effortless use of editor.
- As the editor is not used to an automated environment they wantthe data I\O screens to be user friendly and considerably simple.

### **3.8Other Non Functional Requirements:**

Nonfunctional requirements define system properties and constraints it arises through user needs, because of budget constraints or organizational policies, or due to the external factors such as safety regulations, privacy registration and so on.

Nonfunctional requirements are:

- Security
- Reliability
- Maintainability
- Portability
- Extensibility
- Reusability
- Application Affinity/Compatibility
- Resolution

### **3.9 Quality Attributes:**

The product is targeted towards a wide variety of users such as students, staffs, students cum staff etc. The product must load quickly and must work well on a variety of terminals. It must tolerate wide variety of input possibilities from a user, such as incorrect responses or unforeseen keystrokes.

### **4. Change History:**

200209	Version 1.0 – Initial Release

### **5. Document Approvers:**

SRS for HIC based Multi utility system (including mouse and webcam controller) approved by:

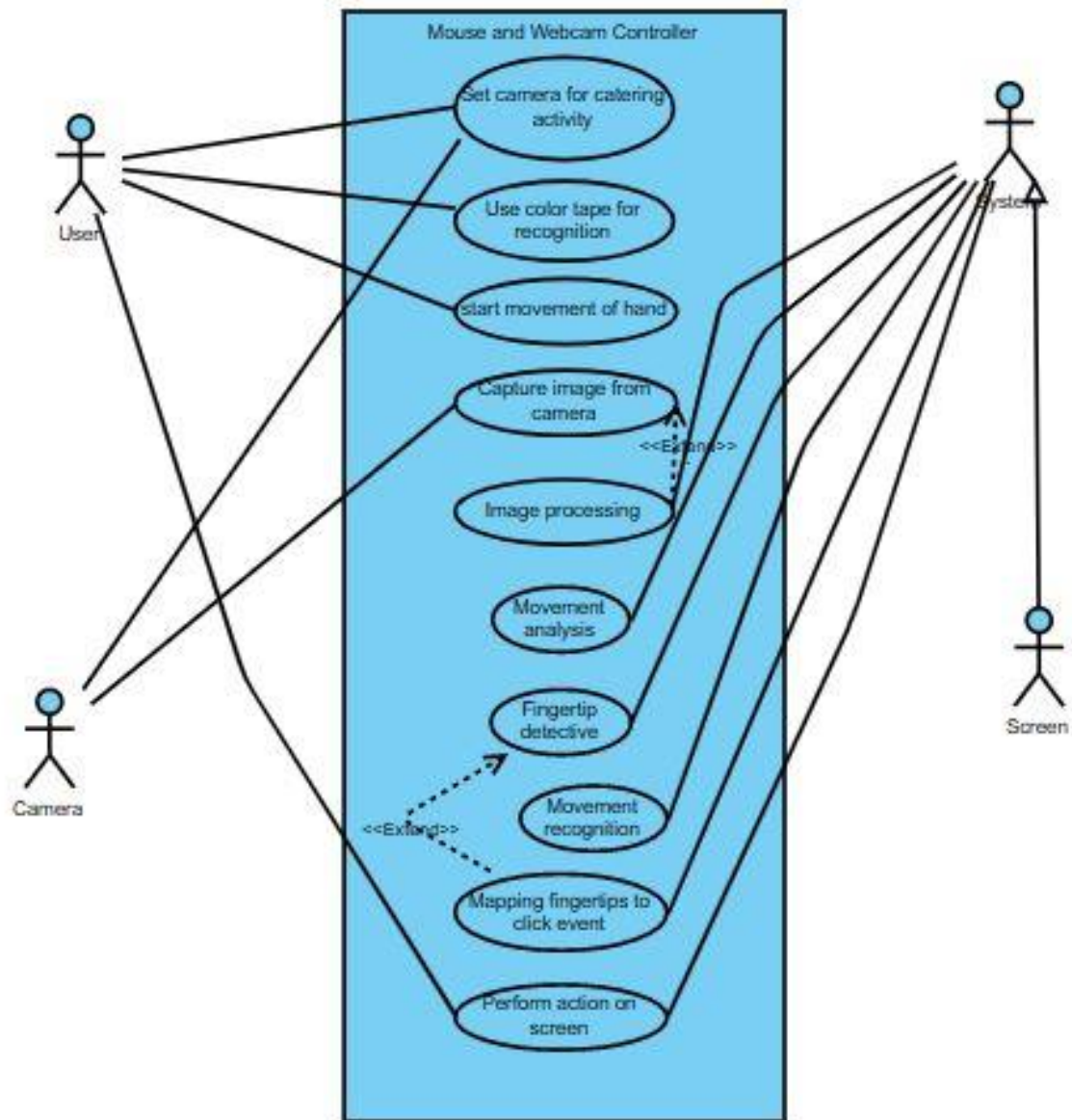
---

(Name)

Designation :

Date:

## USE CASE DIAGRAM:



## USE CASE TEMPLATE :

<b>1.Use Case ID</b>	1
<b>2.Use Case Name</b>	Use color tapes for recognition
<b>3.Created By</b>	Chetanveer Singh,Vansh Sandhir
<b>4.Actors</b>	User, Camera, Screen, System
<b>5.Description</b>	It is the utility that would help to detect colours using grayscale and color tapes.
<b>6.Pre Conditions</b>	Webcam and active system
<b>7.Post Conditions</b>	Monitor or Display
<b>8.Special Requirements</b>	Uninterrupted environment, Bare hands ,Human face or any other object
<b>9.Modification History</b>	28 Feb 2023, Tuesday

<b>1.Use Case ID</b>	2
<b>2.Use Case Name</b>	Clicking operations
<b>3.Created By</b>	Harneet kaur ,Ira Gupta
<b>4.Actors</b>	User,Camera , Screen , System
<b>5.Description</b>	It is the utility that would perform the major function of our proposed model.It would help to replace the traditional mouse by introducing a virtual technology.
<b>6.Pre Conditions</b>	Webcam and an active system
<b>7.Post Conditions</b>	Monitor or Display
<b>8.Special Requirements</b>	Uninterrupted environment, Bare hands
<b>9.Modification History</b>	28 Feb 2023, Tuesday

<b>1.Use Case ID</b>	<b>3</b>
<b>2.Use Case Name</b>	<b>Scrolling and Screenshot</b>
<b>3.Created By</b>	<b>Ira Gupta , Harneet Kaur</b>
<b>4.Actors</b>	<b>User, Camera, Screen, System</b>
<b>5.Description</b>	<b>It is the utility that would take the previous function as consideration but with an advancement .It would be able to take screenshots and perform basic scrolling with system coordinates provided.</b>
<b>6.Pre Conditions</b>	<b>Webcam and active system</b>
<b>7.Post Conditions</b>	<b>Monitor or Display</b>
<b>8.Special Requirements</b>	<b>Uninterrupted environment, Bare hands</b>
<b>9.Modification History</b>	<b>28 Feb 2023, Tuesday</b>



## ACTIVITY DIAGRAM :

