Submitted by:

(101683035) Akshay Sharma, BE Third Year, CSE
(101683033) Abhi Mahajan, BE Third Year, CSE
(101503008) Abhishek Sharma, BE Third Year, CSE

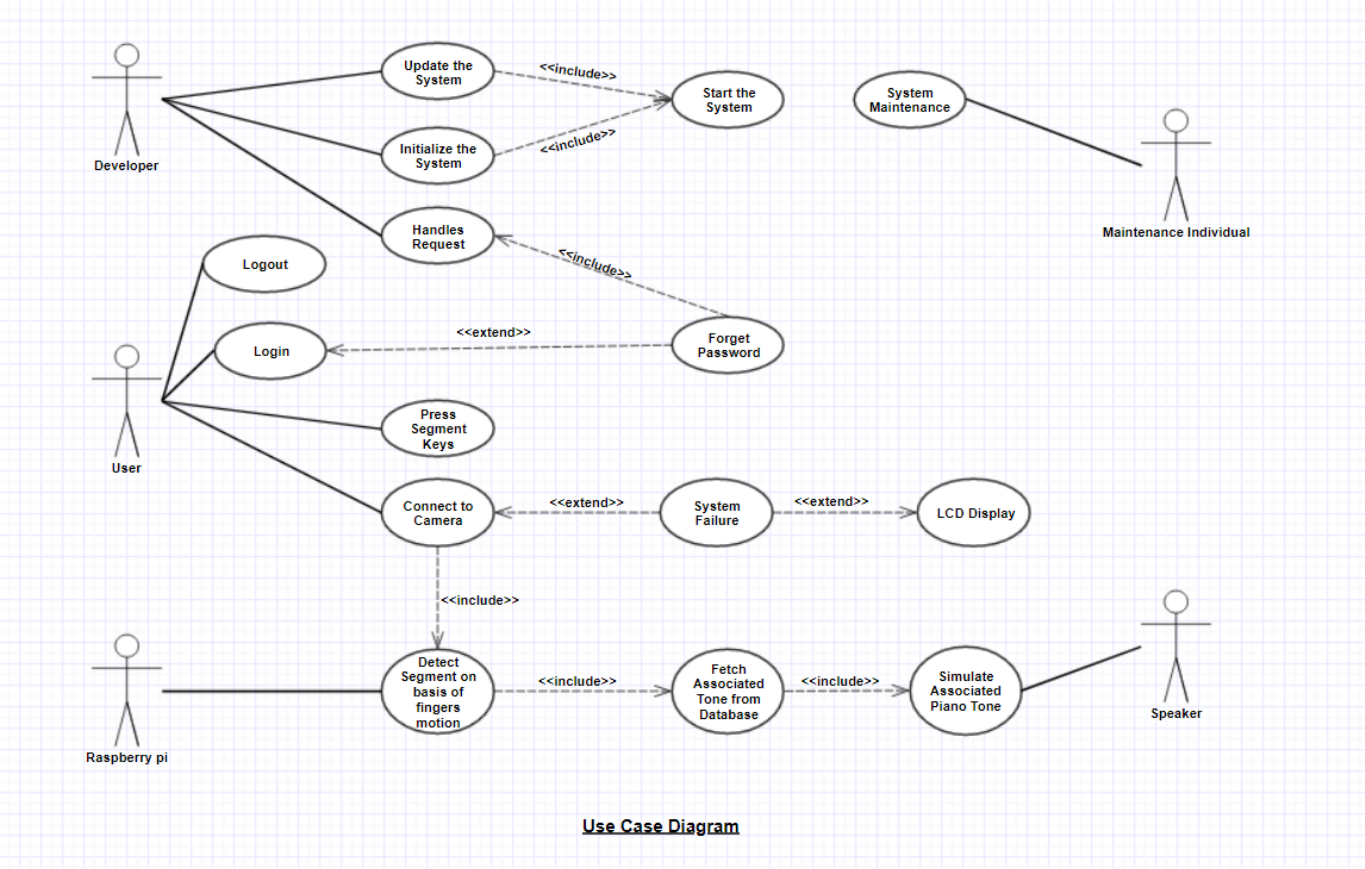
Project Team No. – CPG 84

Under the Mentorship of

Dr. Singara Singh
Assistant Professor
CSED
Thapar Institute of Engineering and Technology, Patiala

COMPUTER SCIENCE ENGINEERING DEPARTMENT
THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY, PATIALA
April, 2018

Use Case Diagram



Use Case Diagram

Use Case Template

Use Case Name:	**Update the System**
Actor:	Developer
Description:	It will update the system as and when required
Preconditions:	The virtual system must be in proper working condition
Postconditions:	The system will be updated to its next version
Priority:	Normal
Frequency of Use:	Low
Includes:	Start the System
Extends:	N/A

Use Case Name:	**Initialize the System**
Actor:	Developer
Description:	The system will enter in its working mode
Preconditions:	N/A
Postconditions:	This will start the system and will enable the user to use it
Priority:	Normal
Frequency of Use:	Low
Includes:	Start the System
Extends:	N/A

Use Case Name:	**Handles Request**
Actor:	Developer
Description:	The system will handle all types of requests made by the user
Preconditions:	The user forgets the password or system updation is required
Postconditions:	The request will be handled successfully
Priority:	High
Frequency of Use:	Low
Includes:	N/A
Extends:	N/A

Use Case Name:	**Login**
Actor:	User
Description:	The user needs to enter his ID so as to use the system
Preconditions:	N/A
Postconditions:	The user will use the system on successful login
Priority:	High
Frequency of Use:	Normal
Includes:	N/A
Extends:	Forget Password

Use Case Name:	**Logout**
Actor:	User
Description:	The user
Preconditions:	The user must be logged in
Postconditions:	The user will be logged out from the system
Priority:	Normal
Frequency of Use:	Normal
Includes:	N/A
Extends:	N/A

Use Case Name:	**Press Segment Keys**
Actor:	User
Description:	The user will press the segments on the plastic sheet to play associated piano tone
Preconditions:	The user must be logged in to the system
Postconditions:	The associated piano tone will be simulated
Priority:	High
Frequency of Use:	High
Includes:	N/A
Extends:	N/A

Use Case Name:	**Connect to Camera**
Actor:	User
Description:	Camera will be mounted above the plastic sheet to detect fingers' motion
Preconditions:	The plastic sheet should be placed properly
Postconditions:	The camera will focus on the plastic sheet and will detect the motion of fingers in real time using Raspberry pi
Priority:	High
Frequency of Use:	Normal
Includes:	Detect Segment on basis of fingers' motion
Extends:	System Failure

Use Case Name:	**Detect Segment on basis of fingers' motion**
Actor:	Raspberry pi
Description:	The pressed segment will be determined from the image captured by the camera in real time
Preconditions:	The camera should be mounted and connected properly
Postconditions:	The associated tone will be fetched from the database and will be played accordingly
Priority:	High
Frequency of Use:	High
Includes:	Fetch associated tone from database
Extends:	System Failure

Use Case Name:	**Simulate associated piano tone**
Actor:	Speaker
Description:	The piano tone will be played based on the detected segment
Preconditions:	The segments must be correctly detected
Postconditions:	N/A
Priority:	High
Frequency of Use:	High
Includes:	N/A
Extends:	N/A

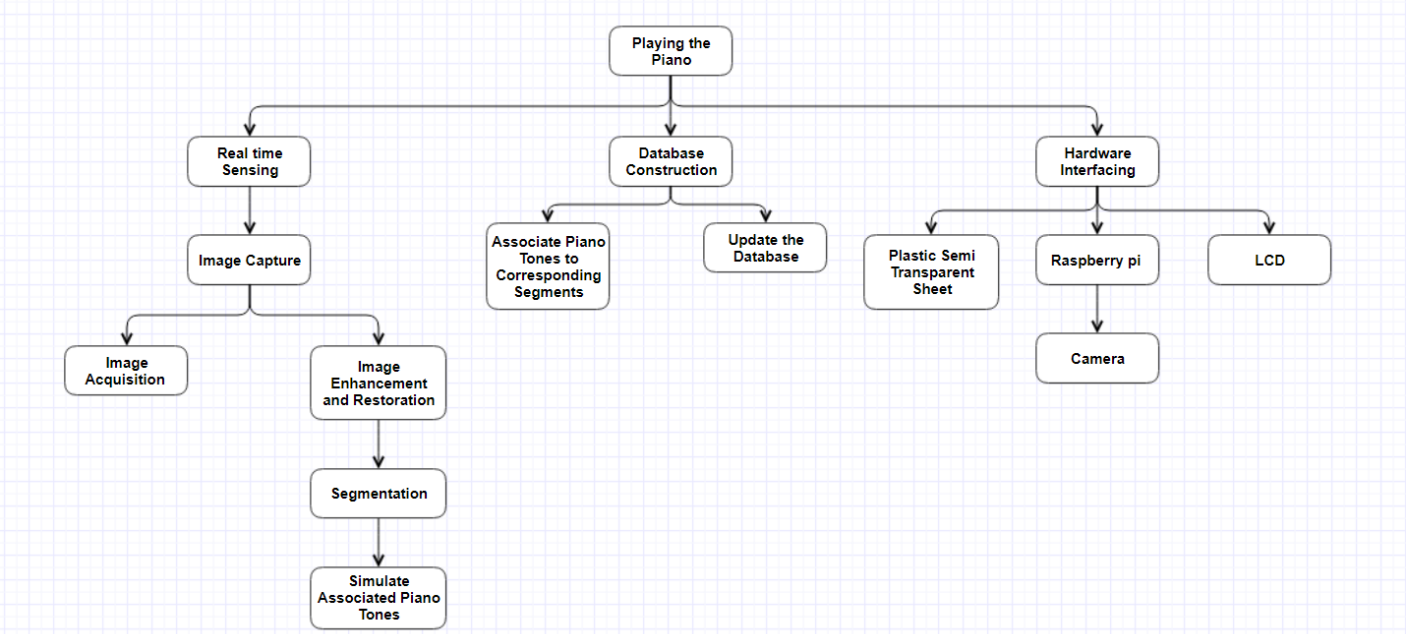
Use Case Name:	System Maintenance
Actor:	Maintenance Individual
Description:	To prevent the malfunctioning of the system, periodic maintenance of hardware is required
Preconditions:	N/A
Postconditions:	The system functions properly
Priority:	Normal
Frequency of Use:	Low
Includes:	N/A
Extends:	N/A

Tasks and Sub-tasks of the Project

- Planning
  - Identify Scope
  - Identify functional requirements
  - Identify non-functional requirements
- Synopsis documentation
- Hardware Interfacing
- Development
  - Develop Application
    - Coding
  - Hand off build
  - Testing
  - Fix bugs
- Performance modification
- Final Documentation
- Execution
- Final Report & Project

Activity Diagram

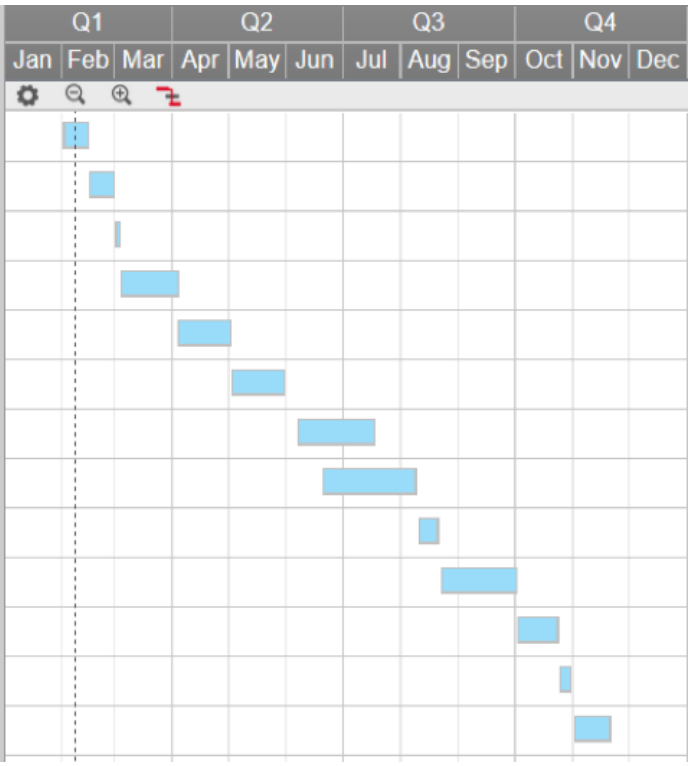
Work Breakdown Structure



Work Breakdown Structure

Scheduling all the tasks in Work Breakdown Structure using Gantt Chart

Task Name	Duration	Start	Finish
Planning of project	10d	02/01/18	02/14/18
Project Research	10d	02/15/18	02/28/18
Synopsis Documentaion	2d	03/01/18	03/02/18
Data sets Collection	23d	03/04/18	04/03/18
Hardware interfacing	21d	04/03/18	05/01/18
Designing	21d	05/02/18	05/30/18
Front end coding	30d	06/06/18	07/17/18
Database Coding	36d	06/20/18	08/08/18
Testing	7d	08/10/18	08/20/18
Performance modification	29d	08/22/18	10/01/18
Documentation	16d	10/02/18	10/23/18
Execution	5d	10/24/18	10/30/18
Final report and Project	14d	11/01/18	11/20/18



Functional Requirements

- The system should do segment detection on the basis of finger's motion.
- The system should simulate associated piano tone based on the segment detected.
- The system shall allow the user to play music as per his/her requirements.
- The system shall give different sound based on the selected Instrument which is listed in a combo box.
- The system should handle the problem of system failure using a LCD display.
- The size of the transparent sheet should accommodate all the notes of the piano.
- The developer can maintain and update the system by reinstalling the current system.

Non-Funcstional Requirements

- Performance: The response time of the system must be fast and smooth.
- Reliability: No error will encounter while user is using the application.
- Ease of use: A good design interface should be constructed with easy control and friendly user interface.
- Accurate: The system should play the correct piano tone by detecting the segment accurately.