GENESIS - Learning Outcome & Mini-project Summary Report

L&T Technology Services

Details

Ver. Rel. No.	Release Date	Prepared By	Reviewed By	To Be Approved	Remarks/Revision Details
1.0	18/02/2022	Akash K 40021156	C Programming on Multiple Platforms		
1.0	18/02/2022	Akash K 40021156	Essentials of Embedded System		
1.0	18/02/2022	Akash K 40021156	Applied SDLC and Software Testing		
1.0	18/02/2022	Akash K 40021156	OOPS with Python		
1.0	18/02/2022	Akash K 40021156	Applied Model Based Design Module		
1.0	18/02/2022	Akash K 40021156	Mastering Microcontrollers with Embedded Driver Development Module		
1.0	18/02/2022	Akash K 40021156	Overview of Automotive Systems		
1.0	18/02/2022	Akash K 40021156	Applied Control Systems and Vehicle Dynamics		
1.0	18/02/2022	Akash K 40021156	Classic Autosar Basic to Intermediate		

Contents

List of Figures

Miniproject – 1: Snake Game [Individual]

Modules:

1. C Programming
2. Git

Requirements

4W's and 1 H's

What:

- Building the Snake game.

Where:

- It has to be easily used by the user.

Why:

- I am developing this simple Snake Game for my mini project.

When:

- It has to be completed on 25th November 2021.

How:

- I am using C programming language for Developing this Simple Snake Game.

LOW Level Requirements:

- Memory: 3 GB.
- File Size: 1 GB.

High Level Requirements:

- Graphics Card: NVIDIA GeForce GTX 780.
- CPU: Intel Core i5-4400E.
- OS: Windows 7.

SWOT Analysis:

- **Strength:**

It can be accessible by the PC with low basic requirements and also without internet access we can play this game.

- **Weakness:**

It will not display a colourful graphical output, due to minimal use of graphics.

- **Opportunity:**

It can be developed with good Graphics to deliver with colourful UI.

- **Threats:**

No software is currently available for updating.

High Level Test Cases

TEST ID	DESCRIPTION	EXPECTED OUTPUT	ACTUAL OUTPUT	END RESULT
HLT_ 1	Compiled without any Error.And accepting the user input.	SUCCESS	SUCCESS	PASS
HLT_ 2	END game after hitting the boundary and body of snake.	SUCCESS	SUCCESS	PASS

Low Level Test Cases

TEST ID	DESCRIPTION	EXPECTED OUTPUT	ACTUAL OUTPUT	END RESULT
LLT_1	Increases the size after eating food.	SUCCESS	SUCCESS	PASS
LLT_ 2	Displaying the Score after the game end.	SUCCESS	SUCCESS	PASS

Design

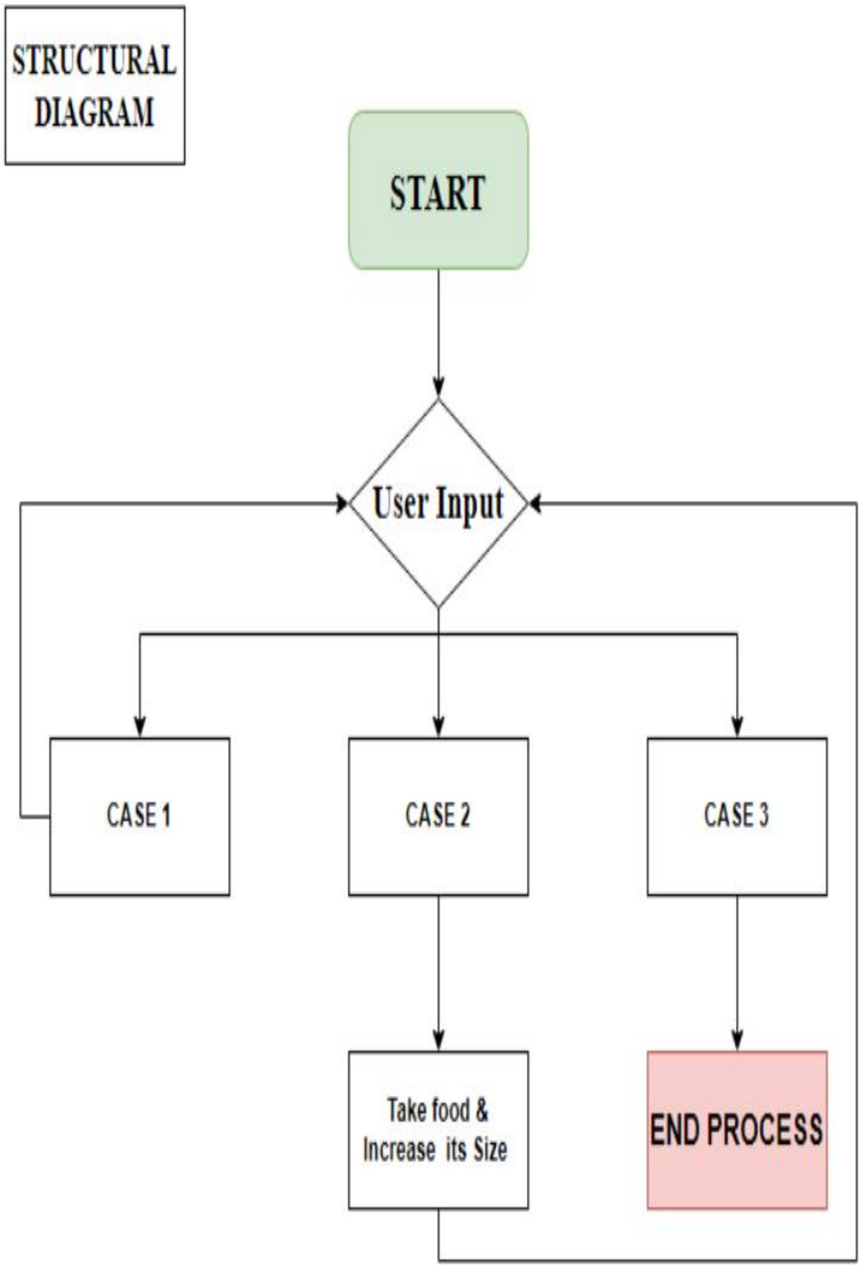


Figure 1Structural Diagram

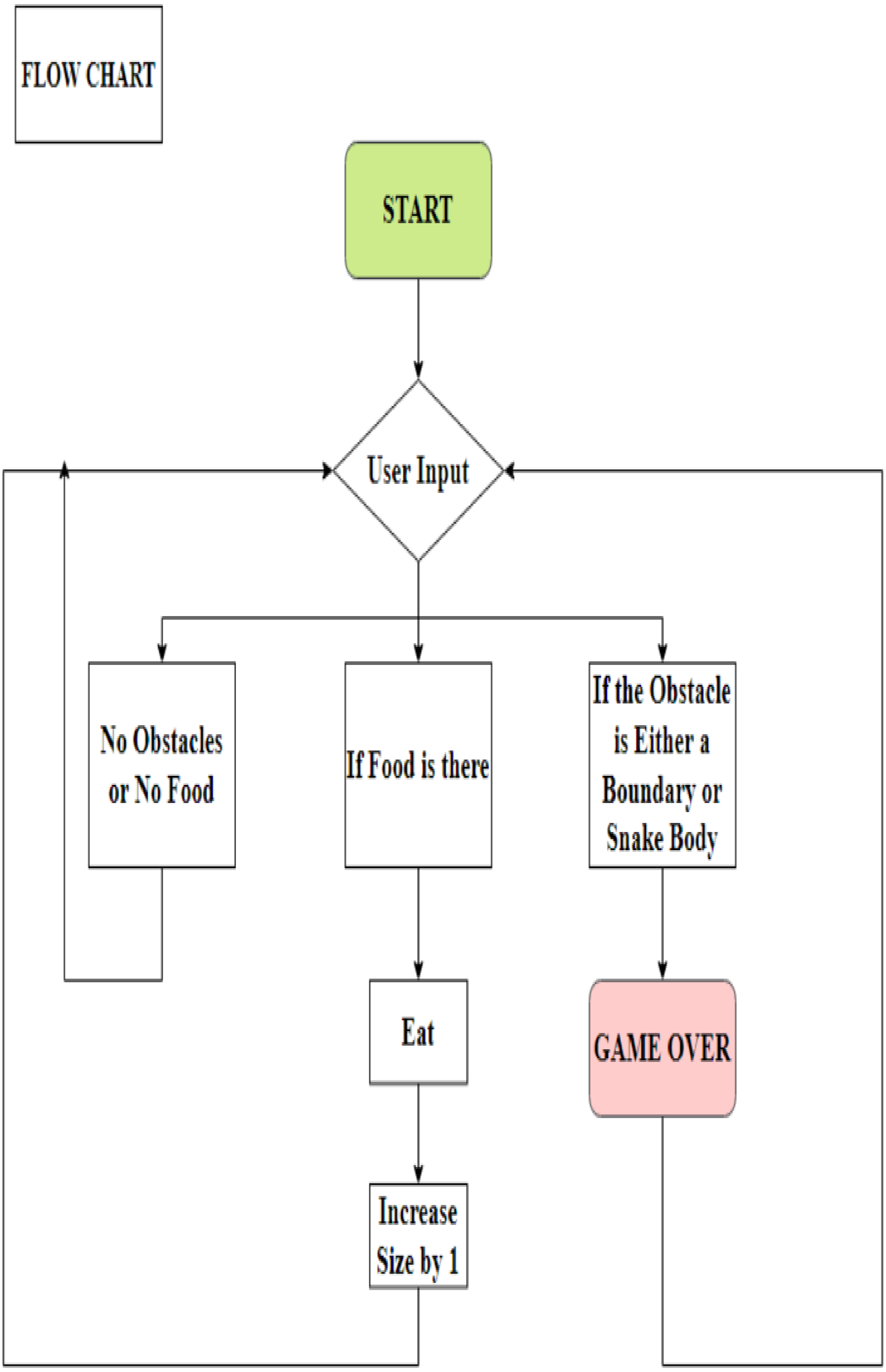
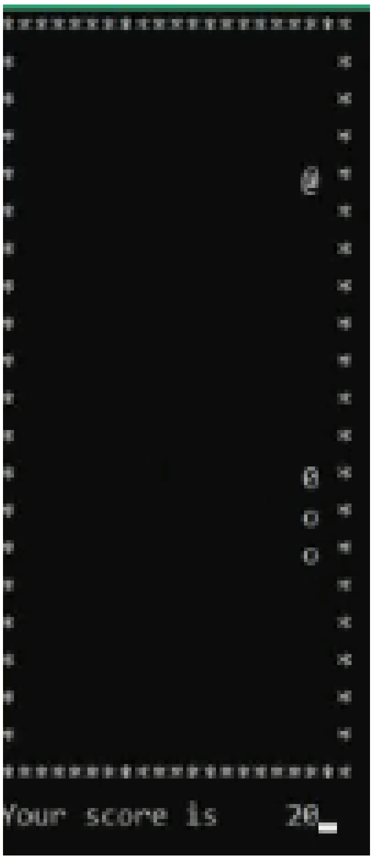


Figure 2 Behavioral Diagram

Output:



```
*********************
*                   *
*                   *
*                   *
*                 @ *
*                   *
*                   *
*                   *
*                   *
*                   *
*                   *
*                   *
*                 0 *
*                 o *
*                 o *
*                   *
*                   *
*                   *
*                   *
*                   *
*********************
Your score is    20_
```

Figure 3 Output

Git Link:

Link: https://github.com/Akash-K29/M1_Game_Snake_Lite

Miniproject 2 – Embedded Water Level Indicator [Individual]

Modules

1. C Programming
2. Embedded System
3. SimulIDE
4. Git

Requirements

4W's and 1 H's

What:

- For calculating water level measurements in water tanks by using conductive sensor by which we can know the measurement of the water in the water tanks.

Where:

- It can be used in water tanks, water in lakes, swaps, and rivers.

Who:

- Every person who wants to calculate the water level measurement by automation by using the conductive sensor can use this.

When:

- It can be used for the water level measurement in the water tanks using the conductive sensor which can kept under the water.

How:

- By using the conductive sensor, we can use to measure water level in water tanks.

SWOT Analysis

- **Strength:** Shows clear indication of water levels and fully automatic.

- **Weakness:** The rust, foul and deteriorate.

- **Opportunities:** Automatic Water level Controller can be used in Hotels, Factories, Homes Apartments, Commercial Complexes, Drainage, etc., It can be fixed for single phase motor, Single Phase Submersibles, Three Phase motors and open well, Bore well and Sump.

- **Threats:** Chances of sensor getting corroded when it is subjected in the salt water.

High Level Requirement:

- **Atmega 328**: The ATMEGA328 is a popular microcontroller due to it being a major component in the Arduino board products. The ATMEGA328 is the 8-bit RISC heart of the Arduino Uno and Nano, with a maximum clock frequency of 20MHz, 32KB program FLASH, and 2KB of RAM.

- **Conductive Sensor:** The sensor has a pair of electrodes, and applies alternating current to them. When the liquid covers the electrodes, it forms a part of an electric circuit, causing current to flow. The sensor measures electric resistance of the liquid, and compares the measured value with a preset value to determine liquid presence.

- **Seven Segment Display:** A seven-segment display is a form of electronic display device for displaying decimal numerals that is an alternative to the more complex dot matrix displays.

Low Level Requirements

- Availability of water in the tank.
- Uninterrupted power supply.
- Embedded C Programming.
- SimulIDE.

Design

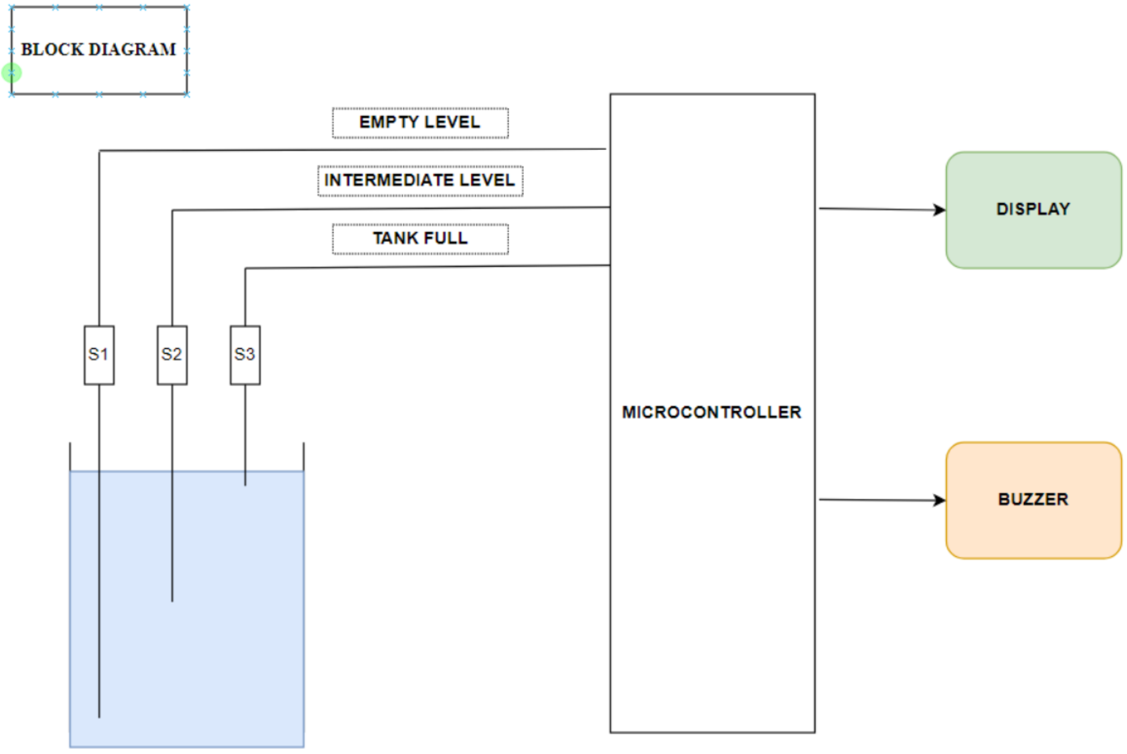


Figure 4 Behavior Diagram

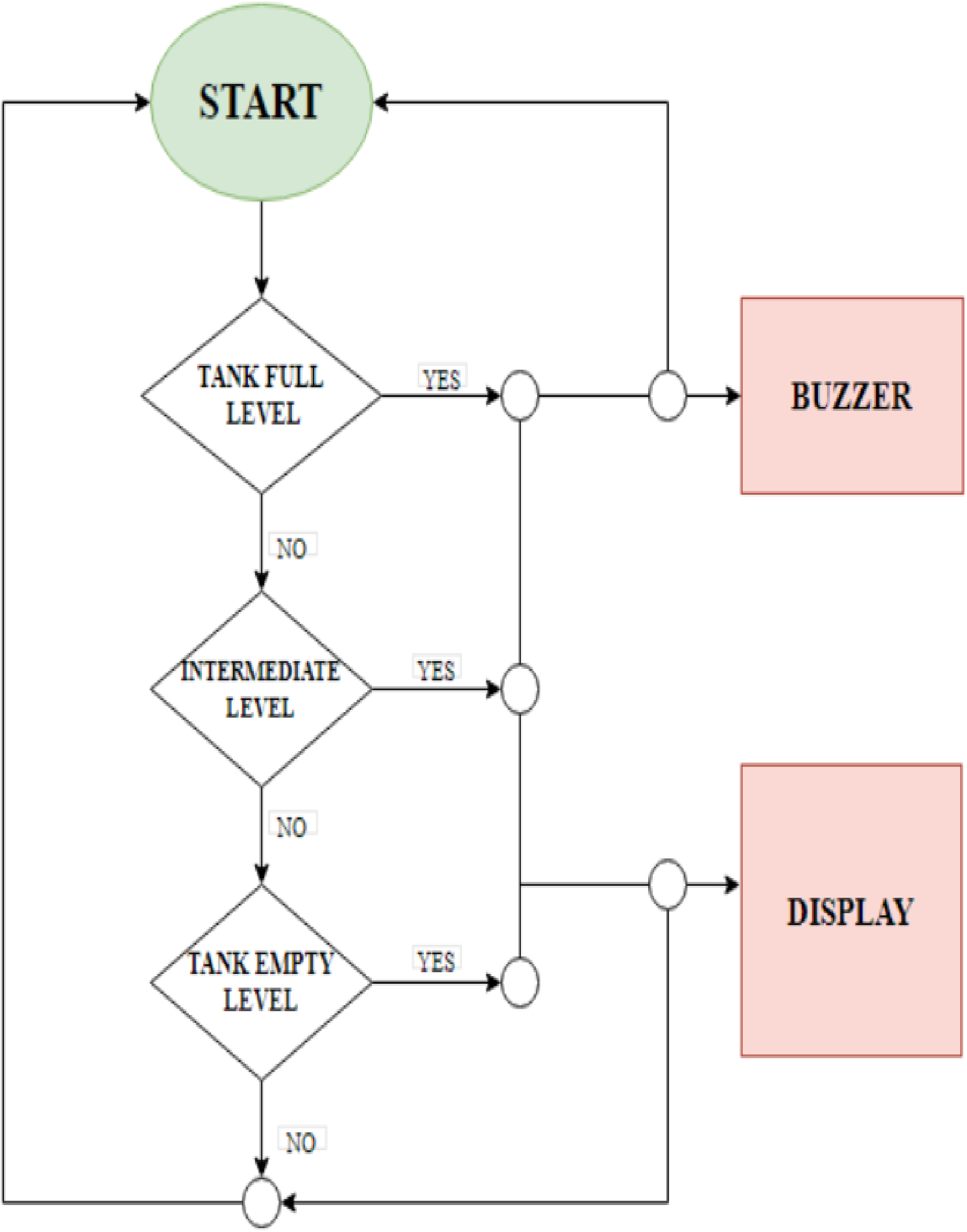


Figure 5Structure Diagram

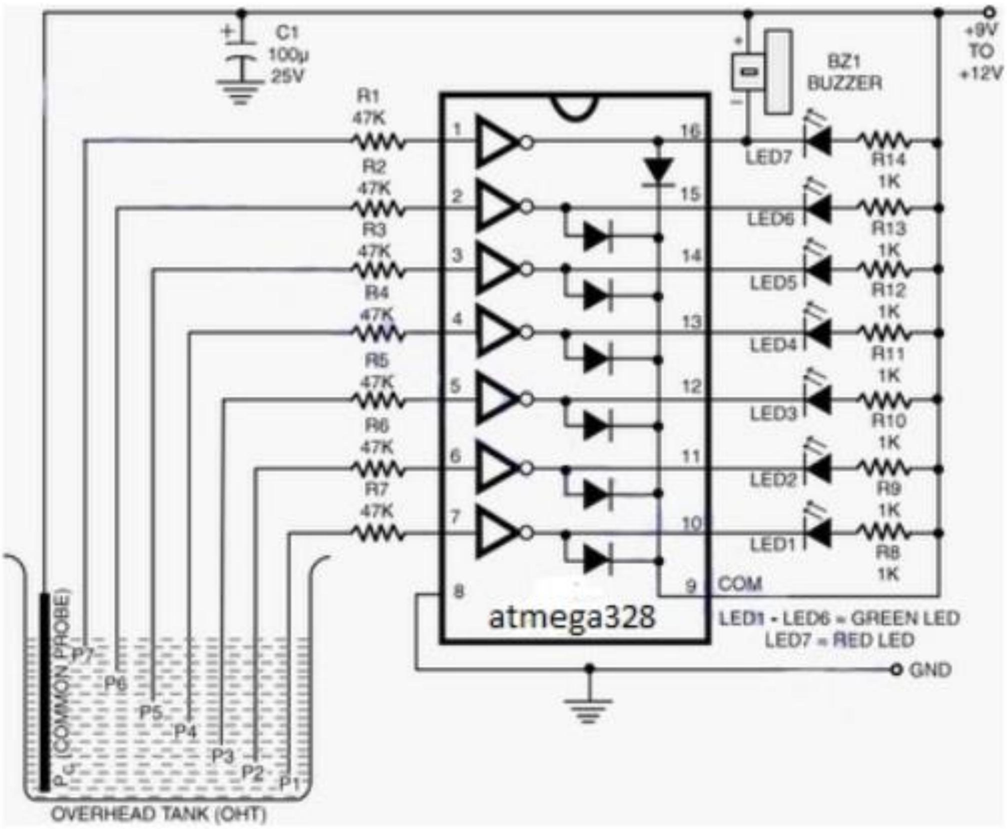


Figure 6Circuit Diagram

Test Plan

High Level Test Plan

TEST ID	DESCRIPTION	EXPECTED OUTPUT	ACTUAL OUTPUT	END RESULT
HLT_ 1	Receiving datas from sensors to the microcontroller	SUCCESS	SUCCESS	PASS
HLT_ 2	Displaying the correct level of the water	SUCCESS	SUCCESS	PASS

Low Level Test Plan

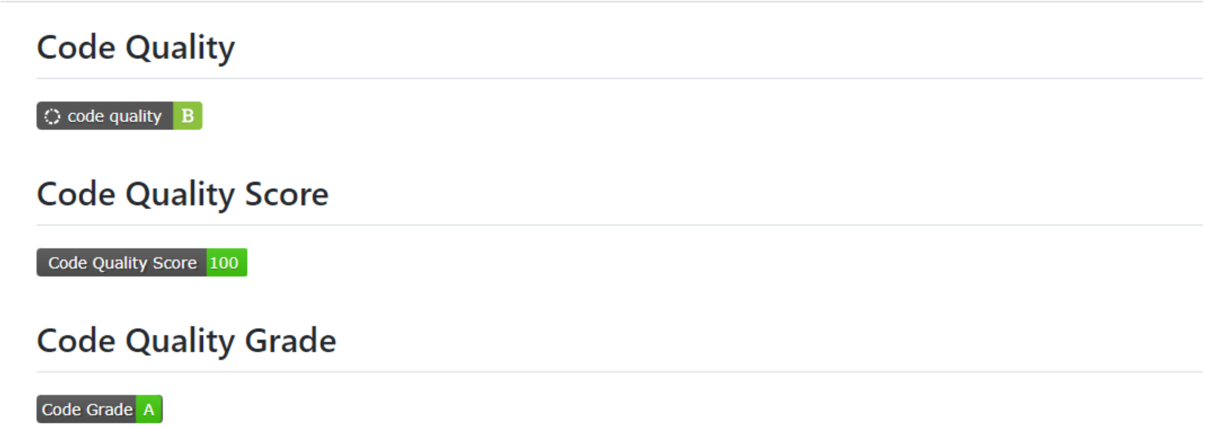
TEST ID	DESCRIPTION	EXPECTED OUTPUT	ACTUAL OUTPUT	END RESULT
LLT_1	turning off the motor when the water level reaches its maximum level	SUCCESS	SUCCESS	PASS
LLT_ 2	Turning on the motor/pump when the water level is minimum level	SUCCESS	SUCCESS	PASS

Implementation and Summary

Git Link:

Link: https://github.com/Akash-K29/M2-Embedded_Water_Level_Indicator

Git Dashboard



Code Quality

code quality B

Code Quality Score

Code Quality Score 100

Code Quality Grade

Code Grade A

Figure 7Git Dashboard

Miniproject 3 – Virtual Costume Advisor [Team]

Modules

1. SDLC
2. Git

Requirements

4W's and 1 H's

Who

- People who want to look good by getting targeted outfit ideas for their body shape.

What

- Calculates the body shape and occasion they are addressing then suggests them the best suitable outfit for their body.

When

- Anytime they want to get themselves dressed well for particular occasions.

Where

- In the Application/system which has this program.

How

- By entering the measurements of the individuals bust size, waist size, high hip size, hip size.

Software Requirements

ID	Description	Platform
HLR_1	Visual Studio code platform	Software
HLR_2	Windows 10 or Linux	Software
HLR_3	Github	Software

High Level Requirements

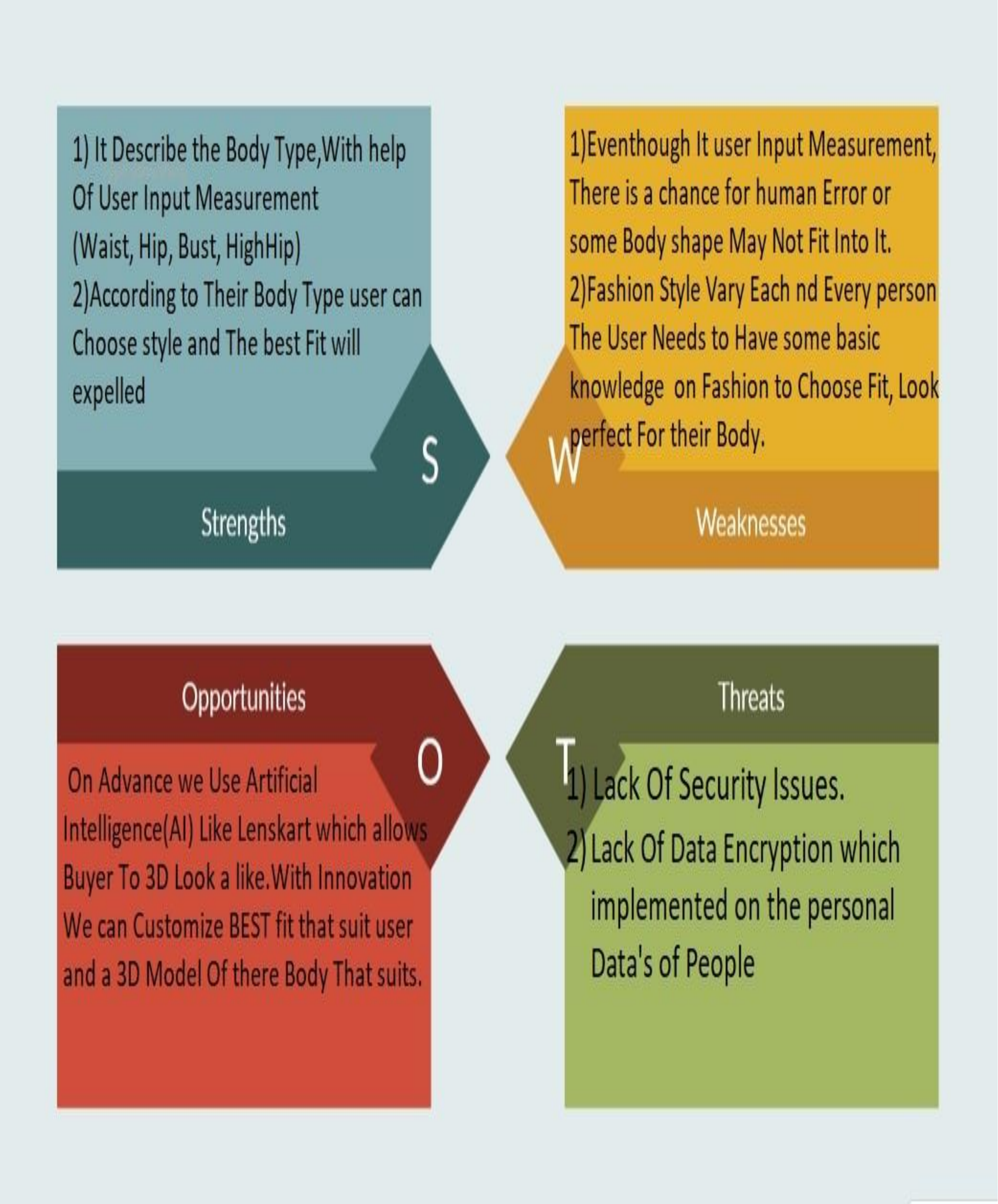
ID	Description	Platform
HLR_1	Getting the measurements from the user	Application
HLR_2	Calculating the body type	Vs code
HLR_3	Getting the choice of outfit type from the user	Application

HLR_4	Getting the choice of listed costume from the user	Application

Low Level Requirements

ID	Description	Platform
LLR_1	The measurements should be properly taken and entered correctly by the user	Application
LLR_2	Coding formula to calculate body type should be accurate	Vs Code
LLR_3	The Choice of outfit type should be properly Chosen and entered correctly by the user	Application
LLR_4	The Choice of costume should be properly taken and given correctly by the user	Application

SWOTAnalysis



1) It Describe the Body Type,With help Of User Input Measurement (Waist, Hip, Bust, HighHip)
2)According to Their Body Type user can Choose style and The best Fit will expelled

1)Eventhough It user Input Measurement, There is a chance for human Error or some Body shape May Not Fit Into It.
2)Fashion Style Vary Each nd Every person The User Needs to Have some basic knowledge on Fashion to Choose Fit, Look perfect For their Body.

On Advance we Use Artificial Intelligence(AI) Like Lenskart which allows Buyer To 3D Look a like.With Innovation We can Customize BEST fit that suit user and a 3D Model Of there Body That suits.

1) Lack Of Security Issues.
2) Lack Of Data Encryption which implemented on the personal Data's of People

Figure 8SWOT Analysis

Design

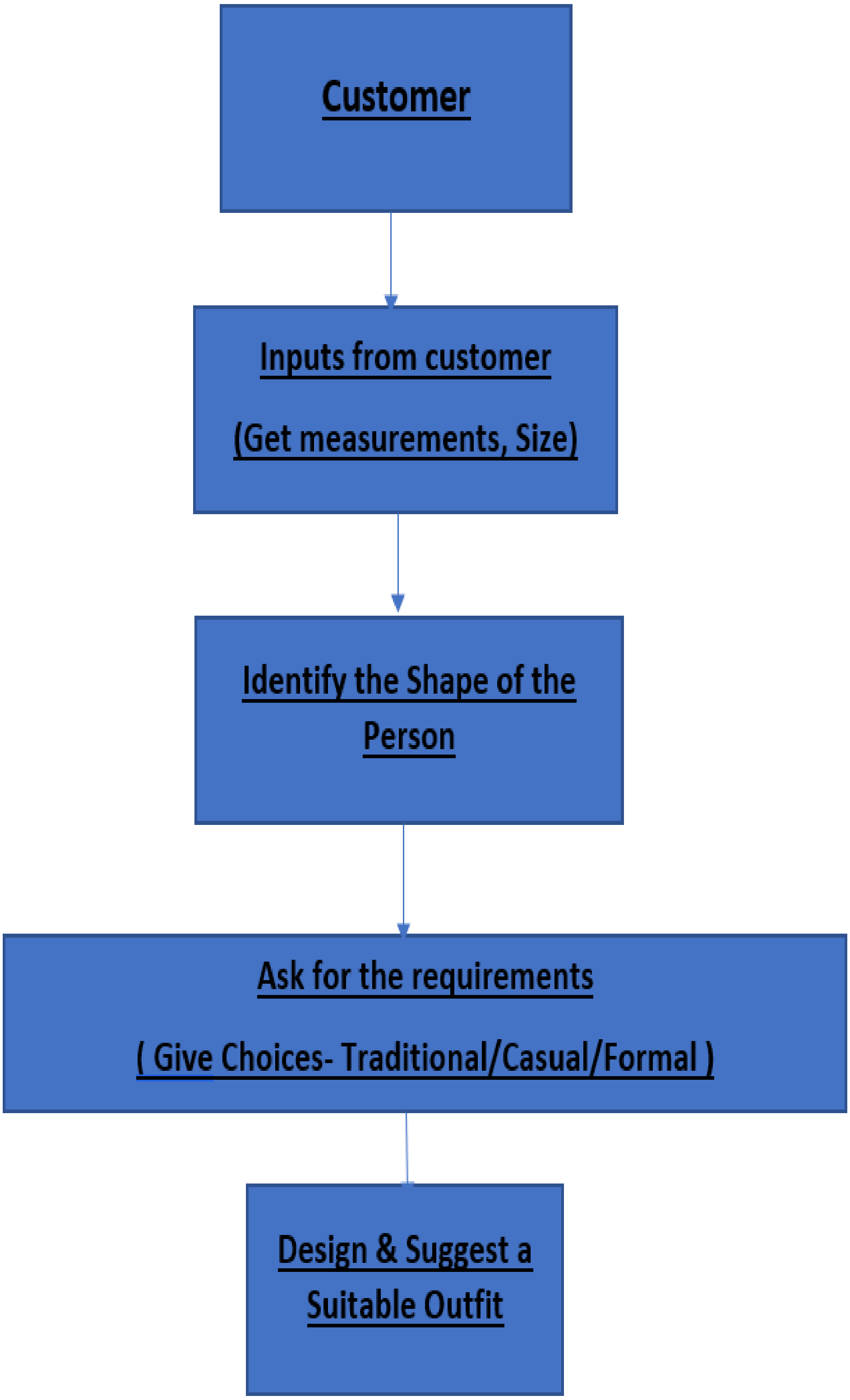

Customer
Inputs from customer
(Get measurements, Size)
Identify the Shape of the Person
Ask for the requirements
(Give Choices- Traditional/Casual/Formal)
Design & Suggest a Suitable Outfit

Figure 9 Low Level Behavior Diagram

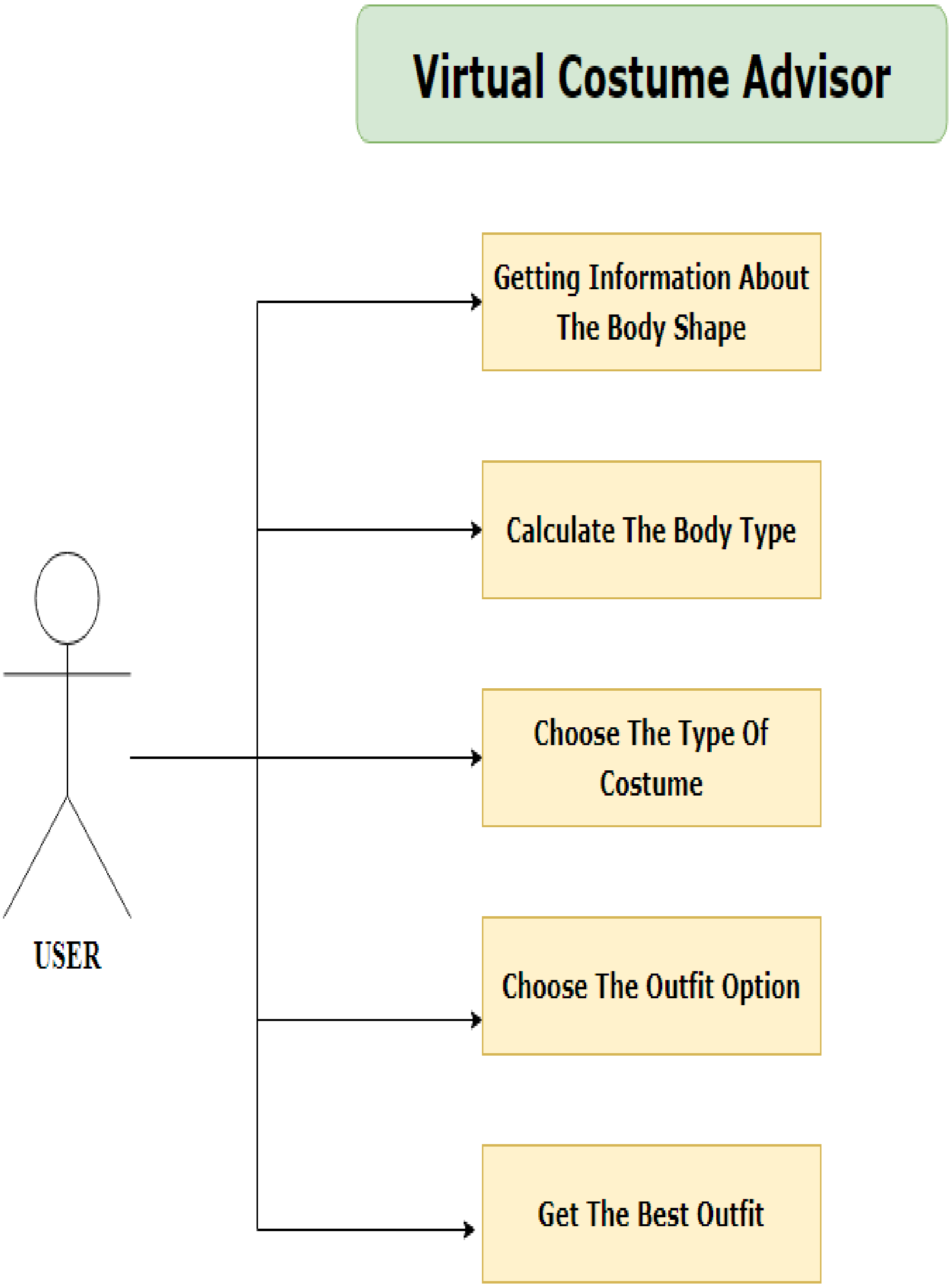


Figure 10High Level Behaviour Diagram

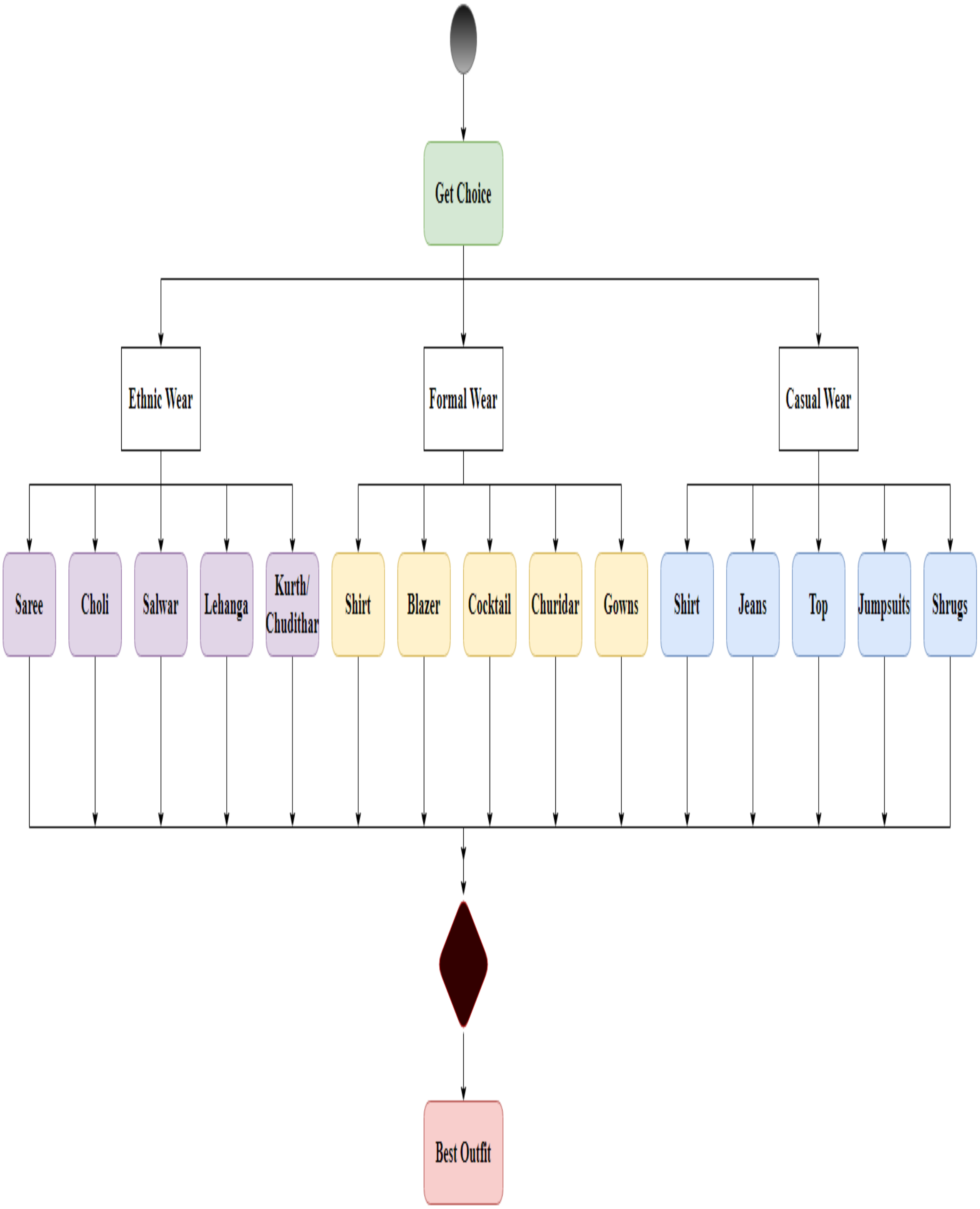


Figure 11Low Level Structure Diagram

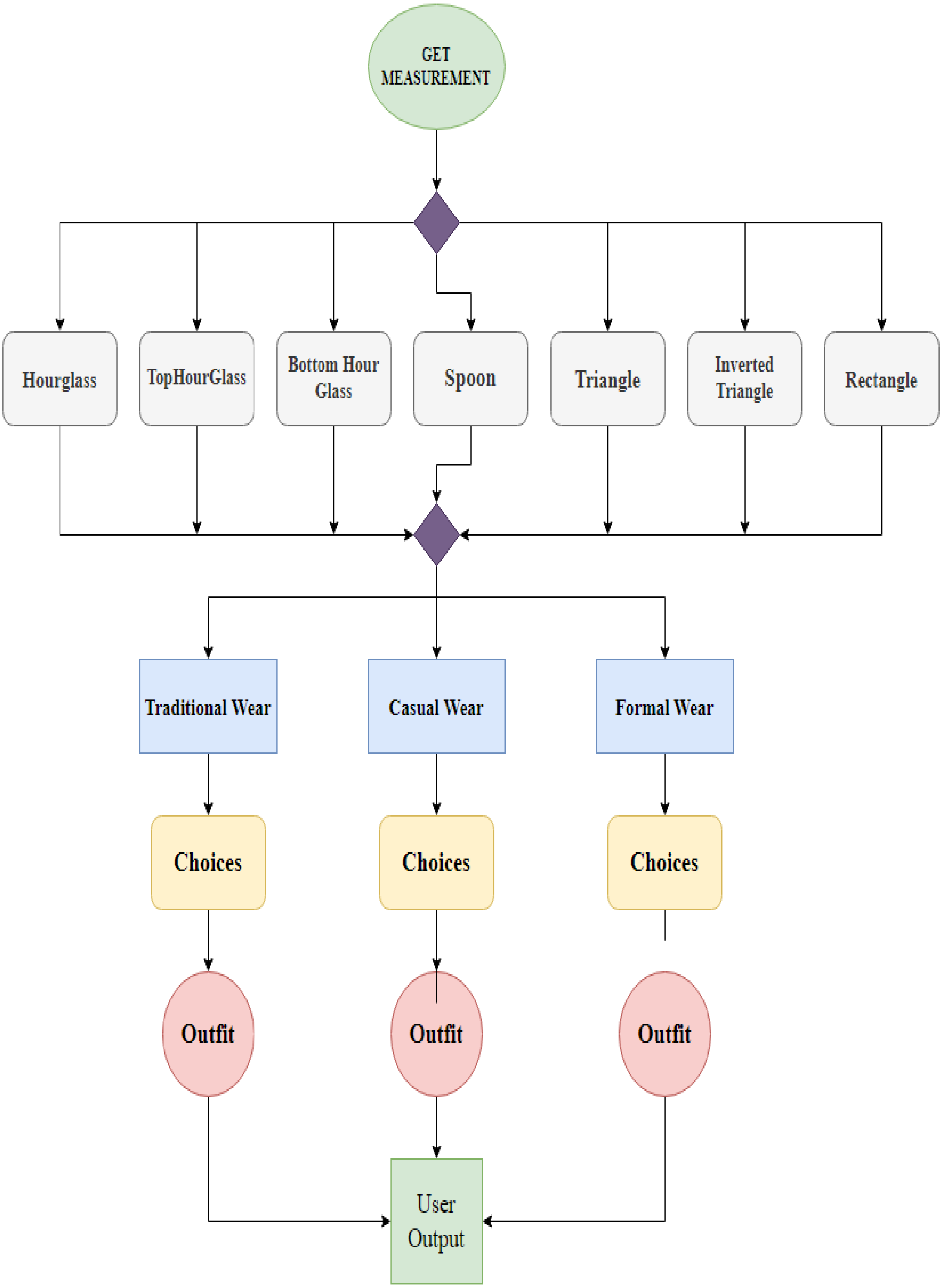


Figure 12High Level Structure Diagram

Test Plan

HIGH LEVEL TEST PLAN:

Test ID	Description	Exp I/P	Exp O/P	Type of Test
H_01	Check and verify all the measurement values are entered	Enter the proper measurements according to the description	Body shape is displayed	Requirement Based
H_02	Check and verify all the measurements entered are in centimeter	Enter the measurements in centimeter	Body shape is displayed	Requirement Based
H_03	Check whether all entered measurements is displayed correctly	Enter the measurements	Bodyshape is displayed	Scenario Based
H_04	Check whether choices of outfit types are displayed correctly	Enter the choice	Another set of choices displayed	Scenario Based
H_05	Check whether choices of costumes are displayed correctly	Enter the choice	The output of desired choice displayed	Scenario Based
H_06	Check whether entered choice of outfit type is correct	Enter the Choice number	Display Choices	Boundary Based
H_07	Check whether entered choice of costumes is correct	Enter the choice number	Display Choices	Boundary Based

Low Level Test Plan

LOW LEVEL TEST PLAN:

Test ID	Description	Exp I/P	Exp O/P	Type of Test
L_01	To check if the measurements give the proper bodyshape	Measurements	Body Type	Requirement Based
L_02	To check if the choices give the proper bodyshape	Choice	The required outfit	Requirement Based
L_03	To check if the calculation is properly done to give proper output	Body Type	Scenario Based	
L_04	To check if all of the four required measurements are entered	Bust, Waist, Highhip, Hip sizes	Display Bodyshape	Scenario Based
L_05	To check if required choice of outfit type is entered	Choices	Display the best outfits	Scenario Based
L_06	To Check if required choice of costume is entered	Choices	Display the best costumes of the desired outfit	Scenario based

Implementation and Summary

Git Link:

Link: https://github.com/GENESIS2021Q1/Applied_SDLC-Dec_Team_50

Miniproject 4 – Calendar Automation [Team]

Modules

1. Python Programming

2. Git

Requirements

4W's and 1 H's

Why:

- It can be used to manage your schedule, events, meetings, and even holiday events.

Where:

- This can be used in our daily lives.

Who:

- Can be used by anyone.

When:

- One can calculate from anywhere.

How:

- The Calendar Automation API allows you to automate your calendar.

High Level Requirements

ID	Feature	MATLAB v0 Status	Python v0 Status
HR01	GUI	Implemented	Implemented
HR02	Master Calender	Implemented	Implemented
HR03	Faculty calender	Implemented	Implemented
HR04	Faculty load sheet	Implemented	Implemented

ID	Feature	MATLAB v0 Status	Python v0 Status
HR05	Showing Available Open Slots based on faculty and modules	Not Available	Not Available
HR06	Output file generated across different computers (windows + linux)	Not Available	Implemented
HR07	Visualizing data to create Meaningful Insights	Not Available	Not Available
HR08	Calculate Individual Faculty Load	Implemented	Implemented

Low Level Requirements

ID	Feature	High Level ID	MATLAB v0 Status	Python v0 Status
LR01	GUI should allow user to login using credentials	HR01	Not Available	Not Available
LR02	Input Files Based on the different Initiatives and Timelines	HR01	Implemented	Not Available
LR03	GUI should get Base Calendar as Input	HR01	Implemented	Implemented
LR04	GUI should get Month and Initiative as Input	HR01	Implemented	Implemented
LR05	GUI should be able to show Conflicts/Warnings	HR01	Implemented	Not Implemented
LR06	Master Calendar: display Month Wise	HR02	Implemented	Implemented
LR07	Master Calendar: display Initiative wise	HR02	Implemented	Not Available
LR08	Master Calendar: Differentiate Initiatives (Color Codes/Numbers)	HR02	Implemented	Implemented
LR09	Master Calendar: Appending	HR02	Implemented	Not Available
LR10	Master Calendar: Course code correction	HR02	Implemented	Not Available

High Level Test Plan

ID	Description	Expected I/P	Expected O/P	Actual O/P	Type Of Test
HLTP_01	Schedule events	User Input	SUCCESS	SUCCESS	Requirement Based
HLTP_02	Review Scheduled events	User Input	SUCCESS	SUCCESS	Requirement Based
HLTP_03	Reminding events	User Input	SUCCESS	SUCCESS	Requirement Based
HLTP_04	Showing available timing Slots based on faculty and modules	User Input	SUCCESS	SUCCESS	Requirement Based
HLTP_05	Output file generation	User Input	SUCCESS	SUCCESS	Requirement Based

Low Level Test Plan

ID	HLTP ID	Description	Expected I/P	Actual O/P
LLTP_01	HLTP_01	User can update their events	SUCCESS	SUCCESS
LLTP_02	HLTP_01	User can review previous schedules	SUCCESS	SUCCESS

ID	HLTP ID	Description	Expected I/P	Actual O/P
LLTP_03	HLTP_02	User will get the remainder of the event	SUCCESS	SUCCESS
LLTP_04	HLTP_04	Multiple event allocated per day will be displayed	SUCCESS	SUCCESS

Implementation and Summary

Git Link:

Link: https://github.com/Ramki17/Calendar_Automation-Genesis21_Team49

Miniproject 5 – Jaguar Project[Team]

Modules

1. Matlab
2. Git

Requirements

We have implemented following features

1. Adaptive Cruise Control System
2. Anti Lock Braking System

Design

This project was implemented using Matlab.

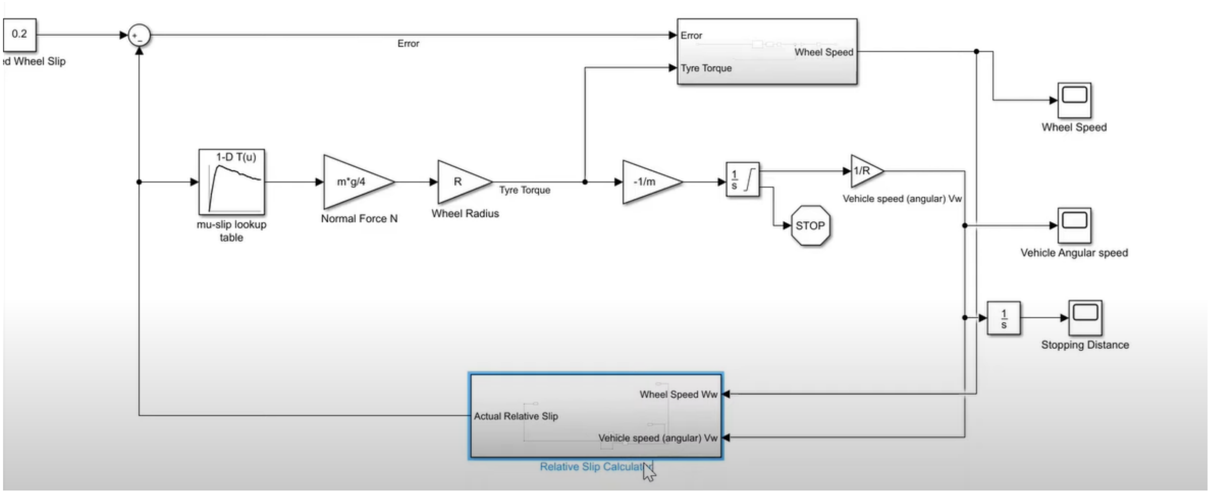


Figure 13 Simulation

Miniproject 6 – Wiper Control[Team]

Modules

1. C Programming
2. STM32

Requirements

4W's and 1H

Who

- Everybody whose drive the car or vehicle

What

- This system is used for cleaning the windshield

When

- A car wiper is a device which is used to remove droplets of rainwater from a windshield of a car.

Where

- At the time of car driving driver need a wiper to clean the windshield

How

- Ignition Key Position at ACC: The Red LED is ON, if the user button is pressed and held for 2 secs
- Wiper ON: Wiper is OFF: On press of the user input, Blue, Green and Orange LEDs come ON one at a time with the set frequency, The frequency changes on every alternate key press, 3 frequency levels with 1, 4 and 8 Hz
- Wiper OFF: Wiper is ON: The LED glow pattern stops on the 4th press; the wiper action starts next press onwards as mentioned in step 2
- Ignition Key Position at Lock: The Red LED is OFF, if the user button is pressed and held for 2 secs

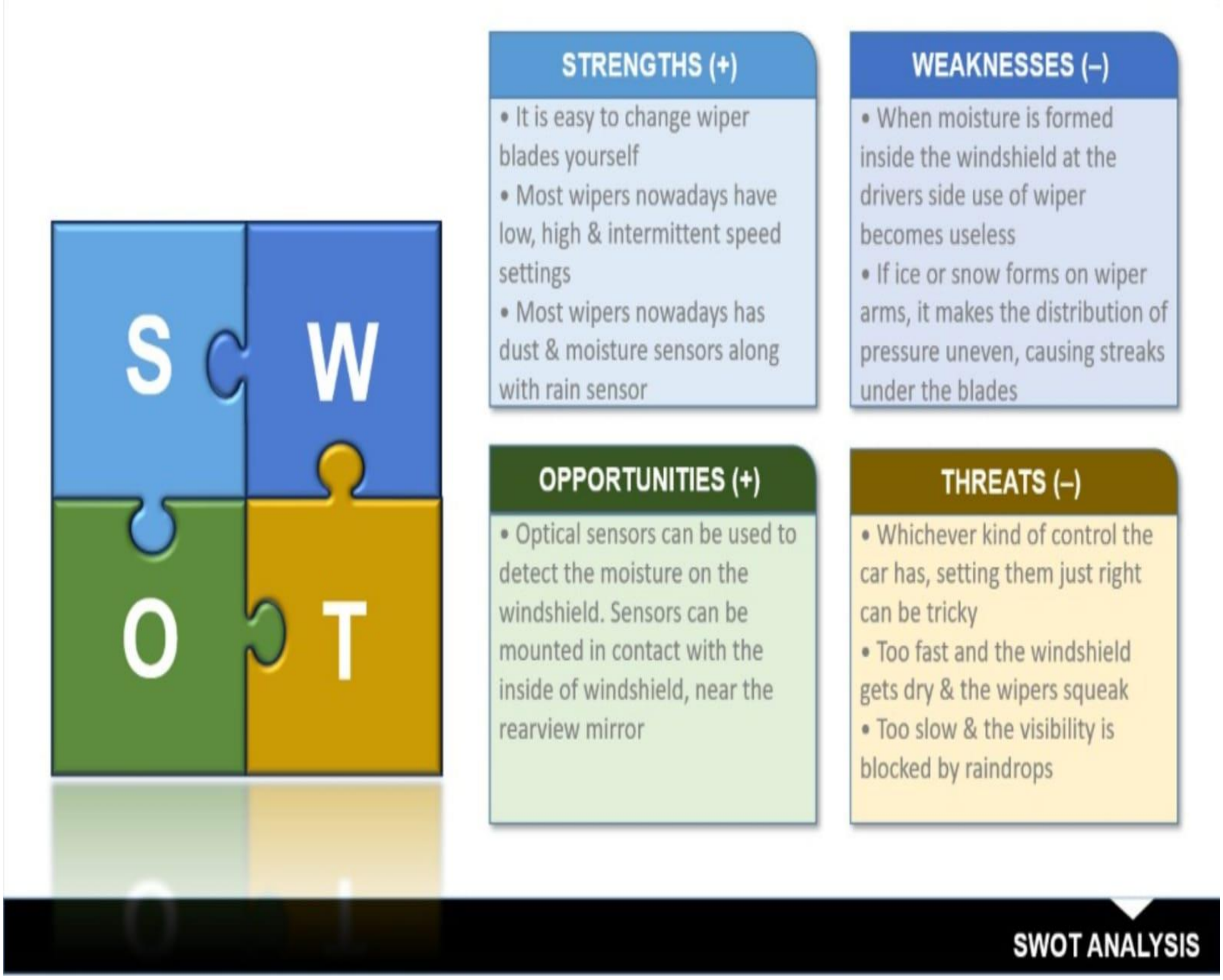
High Level Requirements

HLR	Description	Status
HLR_1	Ignition Key Position to Star	Implemented
HLR_2	Wiper state change OFF to ON	Implemented
HLR_3	Toggling between the LEDs	Implemented
HLR_4	Ignition key Position to Lock	Implemented

Low Level Requirements

LLR	Description	HLR	Status
LLR_1	when Interrupt or event user button is pressed for 2 secs	HLR1	Implemented
LLR_2	Interrupts or events2 after user button continiously with different set frequency	HLR2	Implemented
LLR_3	Interrupts between different frquency(1Hz,4Hz,8Hz)	HLR3	Implemented
LLR_4	Inerrupt or event user button is pressed for 2 sec for lock position	HLR4	Implemented

SWOT Analysis:



STRENGTHS (+)

- It is easy to change wiper blades yourself
- Most wipers nowadays have low, high & intermittent speed settings
- Most wipers nowadays has dust & moisture sensors along with rain sensor

WEAKNESSES (–)

- When moisture is formed inside the windshield at the drivers side use of wiper becomes useless
- If ice or snow forms on wiper arms, it makes the distribution of pressure uneven, causing streaks under the blades

OPPORTUNITIES (+)

- Optical sensors can be used to detect the moisture on the windshield. Sensors can be mounted in contact with the inside of windshield, near the rearview mirror

THREATS (–)

- Whichever kind of control the car has, setting them just right can be tricky
- Too fast and the windshield gets dry & the wipers squeak
- Too slow & the visibility is blocked by raindrops

SWOT ANALYSIS

Figure 14 SWOT Analysis

Design

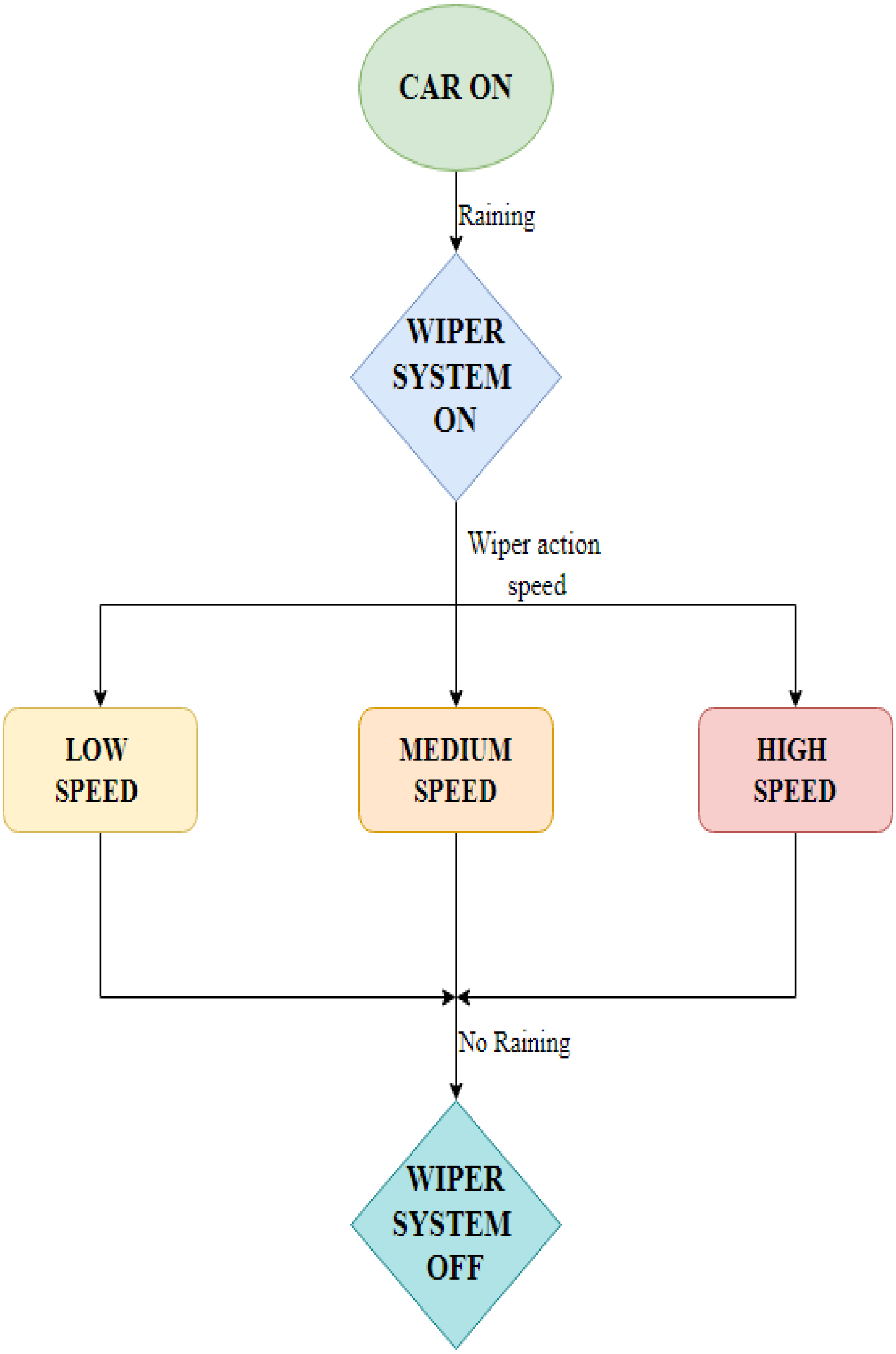


Figure 15 Structure Diagram

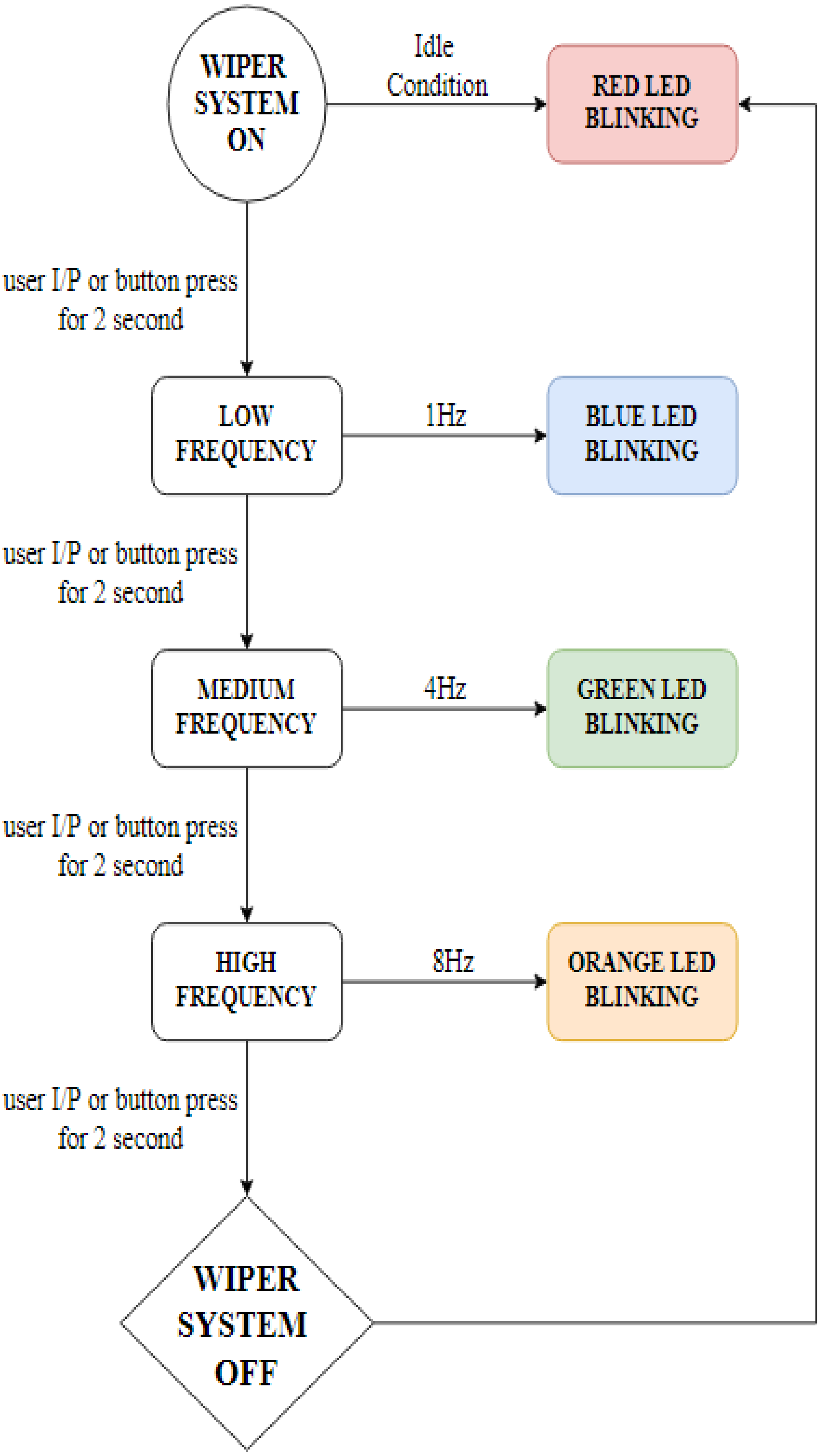


Figure 16Behavior Diagram

Test Plan

High Level Test Plan

Test Id	Description	Expected output	Actual output	Pass/Fail(Result)
H01	The ignition key is pressed for 2 seconds	blink red led	blink red led	Pass
H02	Different frequency shall be set for other leds	succesful	succesful	Pass
H03	The ignition key shall be in off condition pressed for 2 seconds	blink red led	blink red led	Pass

Low Level Test Plan

Test Id	Description	Expected output	Actual output	Pass/Fail(Result)
L01	Frequency shall be set at 1Hz	blink blue led	blink blue led	Pass
L02	Frequency shall be set at 4Hz	blink green led	blink green led	Pass
L03	Frequency shall be set at 8Hz	blink orange led	blink orange led	Pass

Implementation and Summary

Git Link:

Link: https://github.com/GENESIS-2022/MasteringMCU-Team4

Miniproject 7 – Range Rover Project[Team]

Modules

1. Automotive Systems
2. Git

Requirements

Anti-Theft Security System Anti-theft units, which provide protection through the ignition system. Under the hood there is a computer that controls the operation of the engine. There are few Anti-Theft Security System available are

- Tracking systems
- Factory-installed car alarms.
- Immobilizing devices.
- VIN etching.

SNo.	TITLE	Module	Description
SYS_1	Requirement	User(Driver)	Owns The Key Fob whether to turn On /Off Engine.
SYS_2	Requirement	Key Fob	Consist of Button lock, unlock, Hazard, Low beam and rear Door in the car. Which is used to Awake CPU
SYS_3	Requirement	Central Processing Unit	Take control over the module which send data stream to RF. Once it receives data from RKE engine start.
SYS_4	Requirement	Radio Frequency Transmitter	Used to send Radio Frequency Wave.
SYS_5	Requirement	Data Stream	connection-oriented communication, a data stream is a sequence of digitally encoded coherent signals used to transmit or receive.
SYS_6	Requirement	Remote Keyless Entry (RKE)	The remote keyless system's receiver in the car captures the RF signal, extracts it and sends the data stream to the CPU.

Design

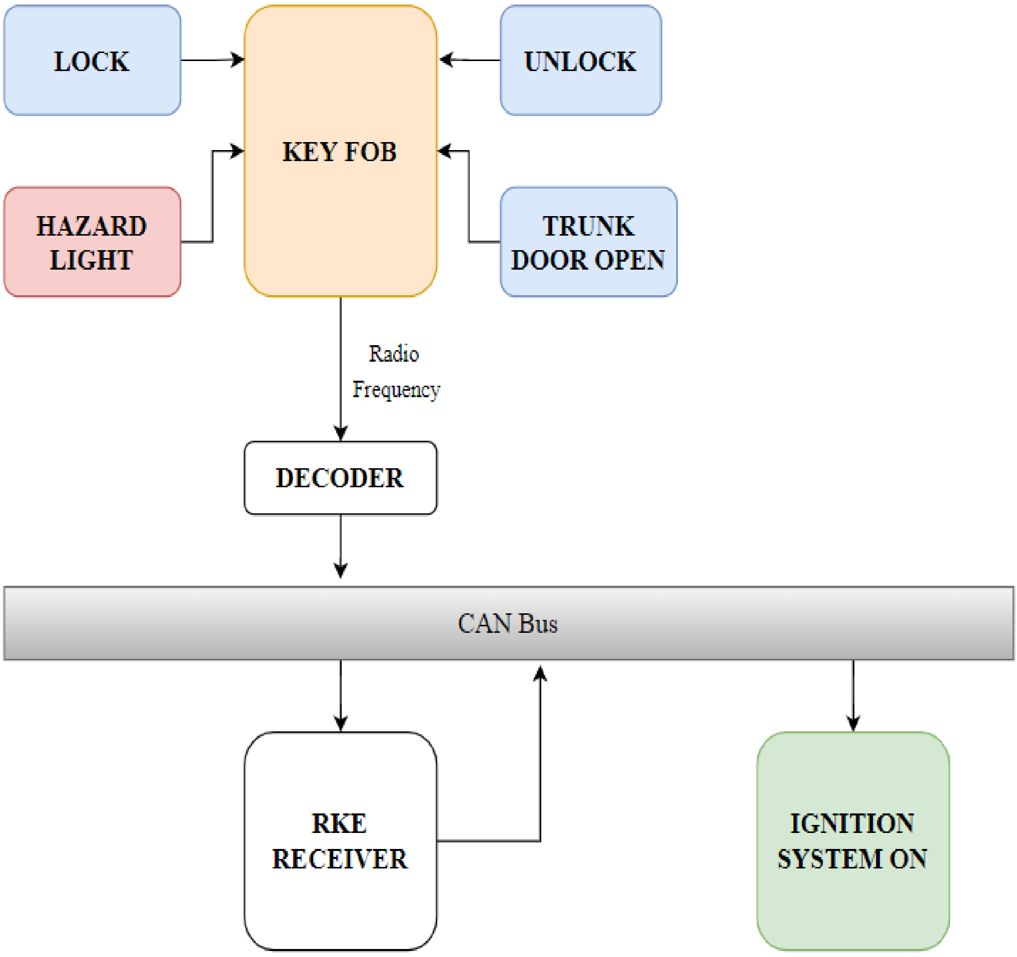


Figure 17Structure Diagram

Implementation and Summary

Git Link:

Link: https://github.com/Ramki17/Automotive-System_RangeRover

Individual Contribution and Highlights

1. Parking System Case Study
2. Source code management using GitHub

Role in Project Team

1. Designer: Done Designing for Project
2. Researcher: Done case study for Parking System

Miniproject 8 – EV Bike[Team]

Modules

1. Matlab
2. Matlab Script

Requirements

Motor Performance:

1. Our Revolt has a Mid Drive IPM motor which can produce 7.2 kW power and 40 Nm torque. We find these figures to be a nice balance of drivability and efficiency.
2. Revolt has an acceleration time from 0 to 60 km/hr of 6.5 seconds.
3. Top speed of our Revolt is 100 km/hr

Battery Performance:

1. We are using a Lithium polymer battery to reduce weight and thereby increase fuel efficiency, performance and handling.
2. A range of 220 km is class leading due to our battery being the biggest at 4.6 kWh.
3. Charging times of our Arrow M1 is higher than the competition at 7.15 hours but we make up for it in the range section.
4. We also offer fast charging.

Braking Performance:

1. Our Revolt also uses combi braking system and use disc brakes for both front and back wheels.
2. Braking performance is on par with the competition.

Wheel Performance:

1. Our Revolt uses Alloy wheels at 12 inches diameter.
2. We use a 90 section, 90 profile tire for a balance between grip, efficiency and ride quality.

Suspension Performance:

1. We use Mono shocks for rear and single fork for front.

Dimensions:

1. Our kerb weight is 110 kg which is just 2 kg heavier than the Ather 450X while having a substantially bigger battery and more powerful motor.
2. Length, Height and Weight are all comparable to the competition.
3. Wheelbase is 1370 mm is the longest in the segment.
4. With a seat height of 782 mm it is accessible for a wide range of people in terms of height.

Implementation and Summary

Submission: Submitted in GEALearn

Individual Contribution and Highlights

1. Done in Matlab Script

Role in Project Team

1. Done Matlab scripting for EV Bike
2. Researcher: Done case study for EV Bike

Miniproject 9 – Anti Thefting System[Individual]

Modules

1. Autosar
2. Git

Requirements

SNo.	TITLE	Module	Description
SYS_1	Requirement	User(Driver)	Owns The Key Fob whether to turn On /Off Engine.
SYS_2	Requirement	Key Fob	Consist of Button lock, unlock, Hazard, Low beam and rear Door in the car. Which is used to Awake CPU
SYS_3	Requirement	Central Processing Unit	Take control over the module which send data stream to RF. Once it receives data from RKE engine start.
SYS_4	Requirement	Radio Frequency Transmitter	Used to send Radio Frequency Wave.
SYS_5	Requirement	Data Stream	connection-oriented communication, a data stream is a sequence of digitally encoded coherent signals used to transmit or receive.
SYS_6	Requirement	Remote Keyless Entry (RKE)	The remote keyless system’s receiver in the car captures the RF signal, extracts it and sends the data stream to the CPU.

Design

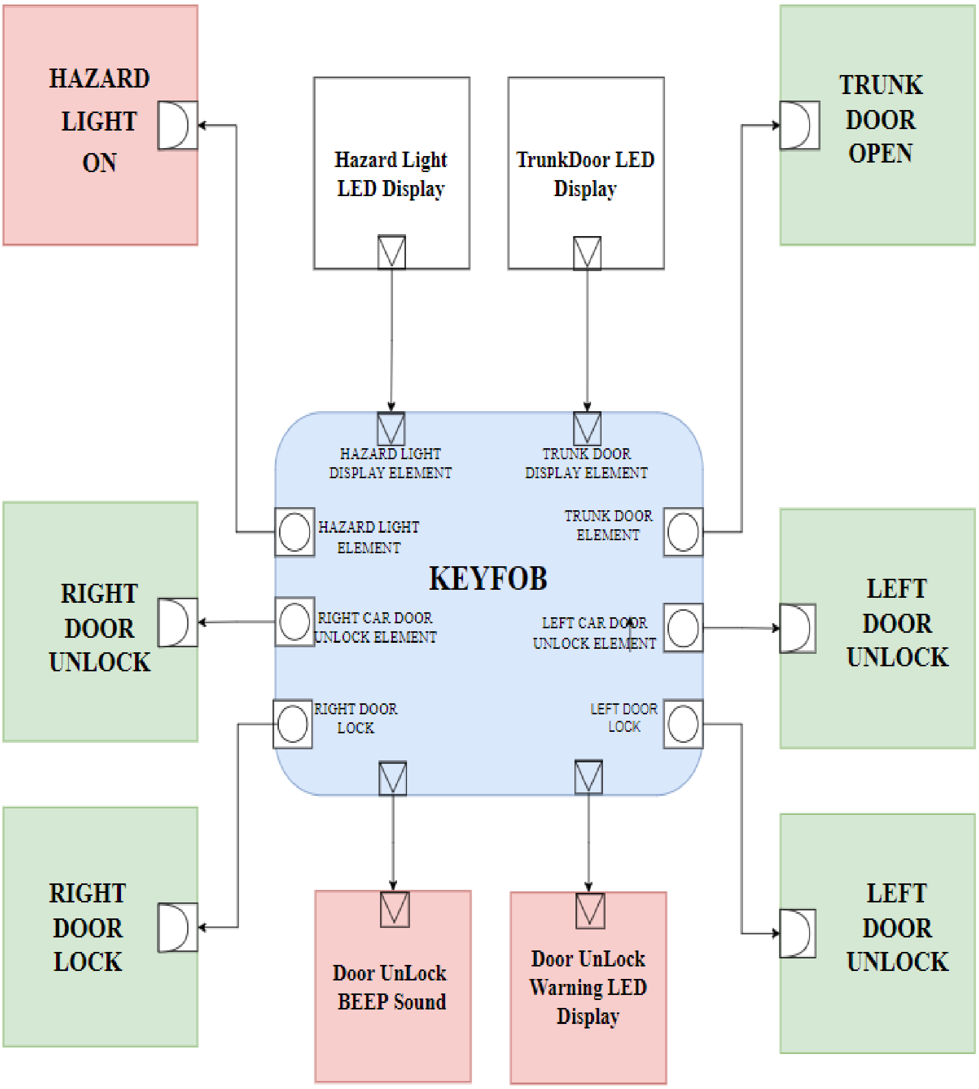


Figure 18 VFB Diagram

Implementation and Summary

Git Link:

Link: https://github.com/Akash-K29/AntiTheftingSystem_40021156_TRN

Individual Contribution and Highlights

1. Anti Theft System Case Study
2. Source code management using GitHub
3. AtomicSwComponent
4. SWCInternalBehavior
5. SWCImplementation