GENESIS - Learning Outcome & Mini-project Summary Report

LTTS GLOBAL ENGINEERING ACADEMY

L&T Technology Services

Details

Ver. Rel. No.	Release Date	Prepared By	Reviewed By	To Be Approved	Remarks/Revision Details
1.0	17/02/2022	Pasumarthi Haritha 40021034			

CONTENTS

List of Figures

Miniproject – 1: CALENDAR [Individual]

Modules: C Programming on Multiple Platforms

INTRODUCTION:

A "CALENDAR" is a system of organizing days. This is done by giving names to periods of time, typically days, weeks, months, and years. A date is the designation of a single, specific day within such a system. A calendar is also a physical record of such a system. A calendar is to be informed about or to agree on a future event and to record an event that has happened. Days may be significant for agricultural, civil, religious, or social reasons.

Calendars play an important role in our daily work to help us stay on task as well as be productive and prioritize. By using them to schedule our daily work we can avoid distractions and get back on track when interrupted.

Feature:

E-mail integration: an electronic mail communication system. This can be tied into the appointment calendar to send reminders and notify the participants of issues arising with scheduled meetings.

Calendar publishing: some calendaring tools allow the user to publish select calendar information on a public link.

Printing: User may print selected schedule. Usually, this feature allows users to select how she wants to have the printout to look (i.e., include comments, subject only, etc.).

Requirements:

4W's and 1'H:

1.Who

Calendar

2.What:

This is done by giving names/events to periods of time, typically days, weeks, months, and years.

3.When:

It is used at specific events/remainder (i.e., Birthday's, Anniversary, Interviews, Meetings etc.)

4. Where:

We can look at mobile phones, google, Desktops, Laptops with different modules.

5.How:

By taking new modules with different updates with latest feature of remainder of events/work/meetings by adding a note.

SWOT- Strengths, and Weakness, Opportunities Threats:

STRENGTHS:

- It keeps us accountable.
- It keeps us realistic. When planning out tasks and activities on the calendar, be realistic.
- It helps us prioritize.
- It keeps us on track.
- It sets boundaries.

WEAKNESSESS:

- Rigidity.
- Misdirected Planning.
- Time consuming.
- Probability in planning.
- False sense of security.
- Expensive.

OPPORTUNITIES:

- A calendar of funding and research opportunities, internships, calls for papers, fellowships, prizes, and job postings that may be of interest to our students, faculty, and alums.

THREATS:

- Data corruption is possible, altering or even deleing events.
- Too many options can make simple event scheduling complicated.
- On PDAs: batteries can fail, leaving your calendar-less.
- Dependent on technology.

Requirements:

High Level Requirements:

ID	Description	Status (Implemented /Future)
Automation	A Calendars can use existing data to make smart suggestions. For example, if you have a monthly team meeting at a local coffee shop, your calendar can automatically suggest exactly when and where that meeting should take place.	Future
Natural language processing	Artificial intelligence is the development of natural language processing. Calendars like Calendly, Doodle and Schedule Once are using natural language so that you can quickly add and edit events.	Future
RSVP tracking and forwarding	Keeping tabs of your events is a key component of time management. But managing who is and isn't attending can be time consuming. Thankfully most online calendars allow you to track responses and RSVPs.	Future

Multiple time zone support	As technology continues to advance, and more companies see the benefit of remote workers, expect that figure to increase dramatically. The catch, however, is that with more people working virtually there's a better chance that you're working with someone in a different time zone. This way you can plan your event accordingly	Future
Integration with other applications	That means if you want to make life much easier, you'll want a calendar that integrates with your existing tools. If you used a scheduling app like Calendar then it should sync with popular calendars like Google, Outlook Calendar, Office Calendar or Apple Calendar.	Future

Low Level Requirements:

ID	Description	Status (Implemented/Future)
The Schedule view	A Schedule view that can show you and your colleagues' calendars side-by-side. It works best with Google Workspace accounts in a company where everyone's calendars are shared. Add your co-workers' calendars from the left sidebar, then select the Day view from the menu in the top left of Google Calendar.	Implemented
Schedule meeting rooms	Calendars are for time, but Google Calendar can also help you organize space. Or spaces, at least. Right beside the Guests tab where you add event attendees, there's an easy to miss Rooms tab, where you can search through your company's available meeting rooms, see what features they have, and add them to an event directly.	Implemented
Track time with a world clock	Google Calendar's scheduling tools help you see when people are busy—but if you're working in different time zones, it might not be quite as obvious when they're sleeping. There's a new Google Calendar tool to help with that: a World Clock that you've likely already noticed in the screenshots.	Implemented
See more with a year view	Google Calendar has a year view. You probably didn't notice they added it, but they did. You can use it. Click the view menu in the top right and select Year to see the entire year briefly. Dates with events aren't highlighted at all, but you can click on a date to see everything scheduled in a popover. Or double-click a date to open it in Day view.	Implemented
Weather is missing	One Google Calendar feature that disappeared in the past few years is weather forecasts.	Implemented

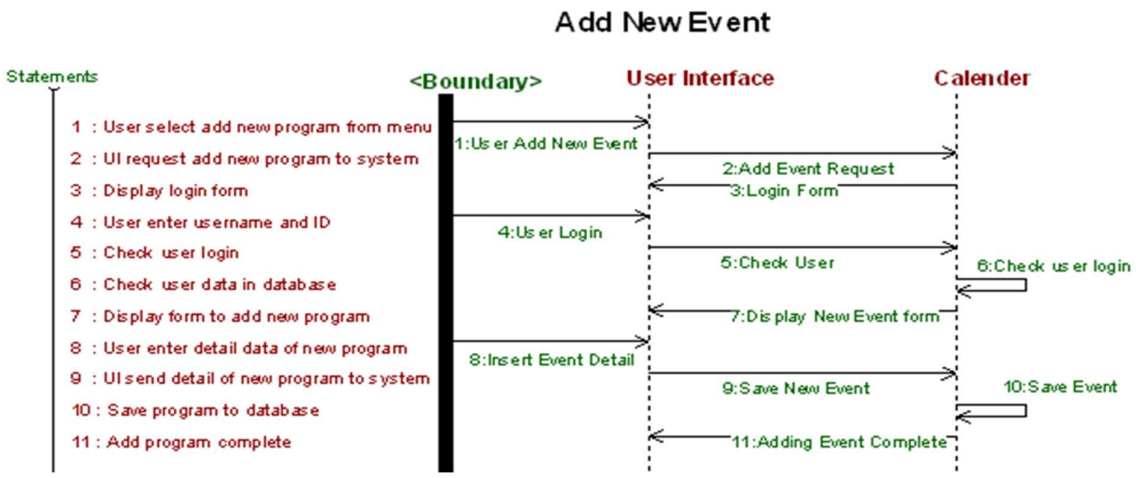


Figure-1 Behaviour Diagram

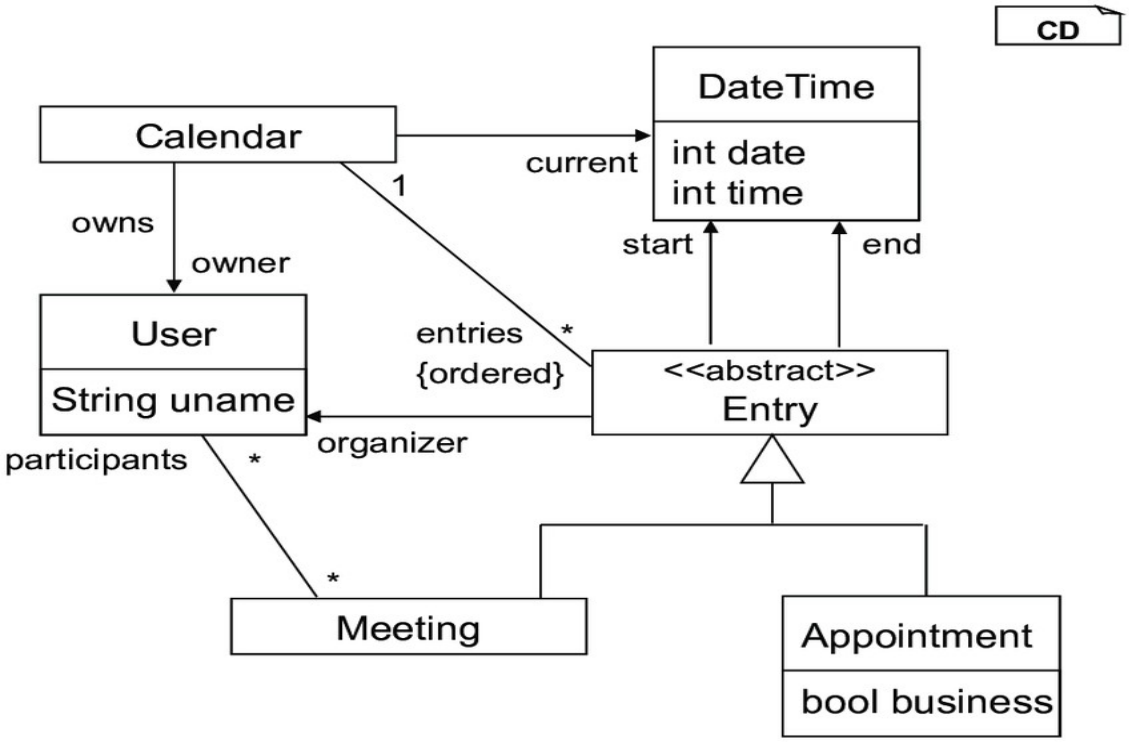


Figure 2 Structure Diagram

High level test plan:

Test ID	Description	Exp I/P	Exp O/P	Actual Out	Type Of Test
H_01	User wants to add an item to the calendar	Enter item/event	Saved successfully	Saved successfully	Requirement based
H_02	User wants to include and invite an attendee to their calendar item.	Sent Invitation	Sent successfully	Sent successfully	Scenario based
H_03	User wants to upload an attachment to their calendar item from desktop	Upload Attachment	Attached Successfully	Attached Successfully	Boundary based

Low level test plan:

Test ID	Description	Exp IN	Exp OUT	Actual Out	Type Of Test
L_01	User wants to make an amendment to their calendar item.	Amend calendar	Amended successfully	Amended successfully	Requirement based
L_02	User wants to cancel their calendar item.	To cancel the item	Cancelled successfully	Cancelled successfully	Scenario based
L_03	User wants to share an online video meeting link to their invite.	Sharing online meet link	Shared successfully	Shared successfully	Boundary based

Implementation and Summary

Git Link:

Link: https://github.com/pasumarthiharitha/M1_APP_CALENDER.git

Git Dashboard:

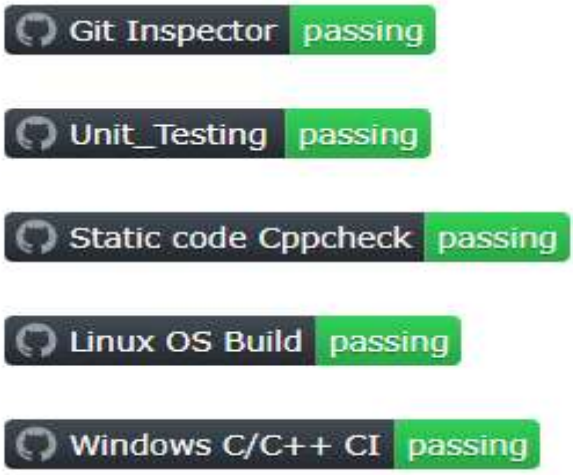


Figure 3 Git Dashboard

Git Inspector Summary:

```
23
24  Author                    Commits    Insertions      Deletions    % of changes
25  pasumarthiharitha              78          5508            723          100.00
26
27  Below are the number of rows from each author that have survived and are still
28  intact in the current revision:
29
30  Author                    Rows      Stability          Age       % in comments
31  pasumarthiharitha         5622          102.1          0.2                7.60
32
```

Figure 4 Git Inspector Summary

CERTIFICATIONS DONE IN MODULE:

- SOLO-Learn Certification
- Linux Certification
- GitHub Learning Certification

Miniproject 2 – Embedded Home Automation [Individual]

Module: Essentials of Embedded System

Tools:

- C Programming
- Embedded System
- Simul IDE
- Git

Requirements:

4W's and 1 H's:

Why:

The project main aim is to make a robust system which can automatically switch on and off the fan according to requirement.

Where:

As in the super coach buses there are separated compartment for every passenger's so according to their need the system can be deployed.

Who:

Any user who is travelling in the Home can use this system.

When:

To improve the quality of life for humans while travelling we can use this system.

How:

By developing an embedded system which is user friendly and can be implemented without difficulty.

Research:

Automation is a necessity in our day-to-day life because it not only seeks to improve the quality of life for humans at both home and work.

- It allows the distribution of both quality products and services to be made available at faster rates, reduces time and human error.
- These days the technology is developing at very high-speed innovation and advancements is there in all the sectors.
- The Temperature measurement system for Bus can maintain the proper temperature inside.

- The main aim of this project is to sense the temperature inside the HOME using temperature sensor (LM 35) and it will give the signal to fan which is present above the passenger's seat which will get switched ONN and OFF according to requirement.
- The fan will get on and off based on the temperature inside the HOME at that moment. for Example: - if the temperature is high inside the HOME so the fan will switch on and if the temperature is low so fan will switch off Based on this our controller will on and off the fan accordingly.
- The temperature sensor (LM35) will sense the heat inside the bus and an LCD display will show the temperature. In our project we have used ATmega328 microcontroller along with temperature sensor, Push button, LED, and LCD display etc.

Features:

- The System will sense the temperature and switch the fan ONN and Off Accordingly.
- Low cost and robust system.
- Modular Approach.

SWOT- Strengths, and Weakness, Opportunities Threats:

Strengths:
User Friendly and easy to use. Easy to alter the temperature inside the bus. Modular Approach, Low cost, and Robust system.

Weakness:
Speed control is independent of individual preference it will either on or off.

Opportunities:
Save energy by switching off in low temperature and it is a modern way of approach.

Threat:
Micro controller is the heart of the circuit, if controller is damaged the whole system will be interrupted.

High Level Requirements:

ID	Description	Status
HLR_1	User shall be able to see the present Temperature	Implemented
HLR_2	Fan should automatically switch off when temperature is low	Implemented
HLR_3	Fan should automatically switch on when temperature is high	Implemented

Low Level Requirements:

ID	Description	Status
LLR_1	Temperature should be display on LCD Screen	Implemented

Design:

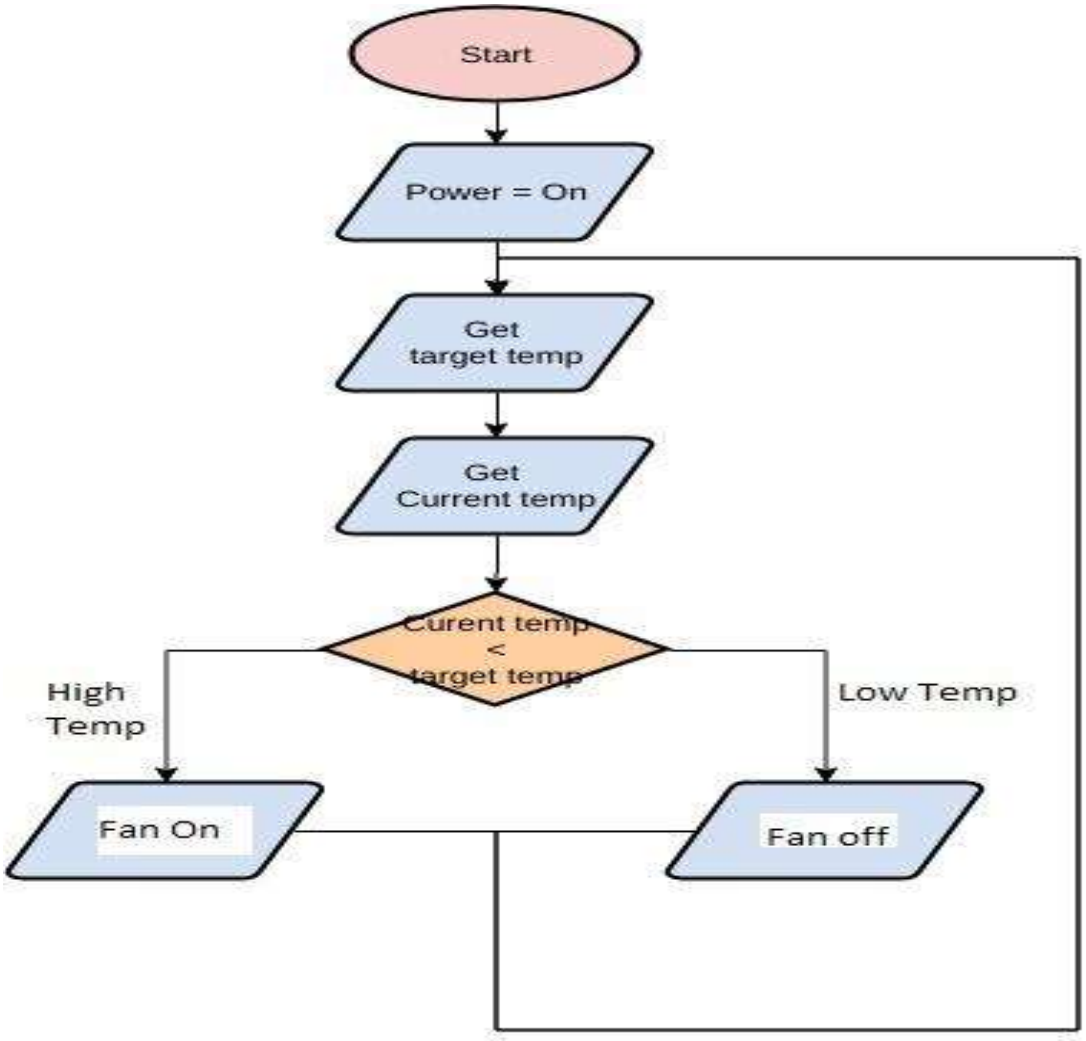


Figure 5 Behaviour Diagram

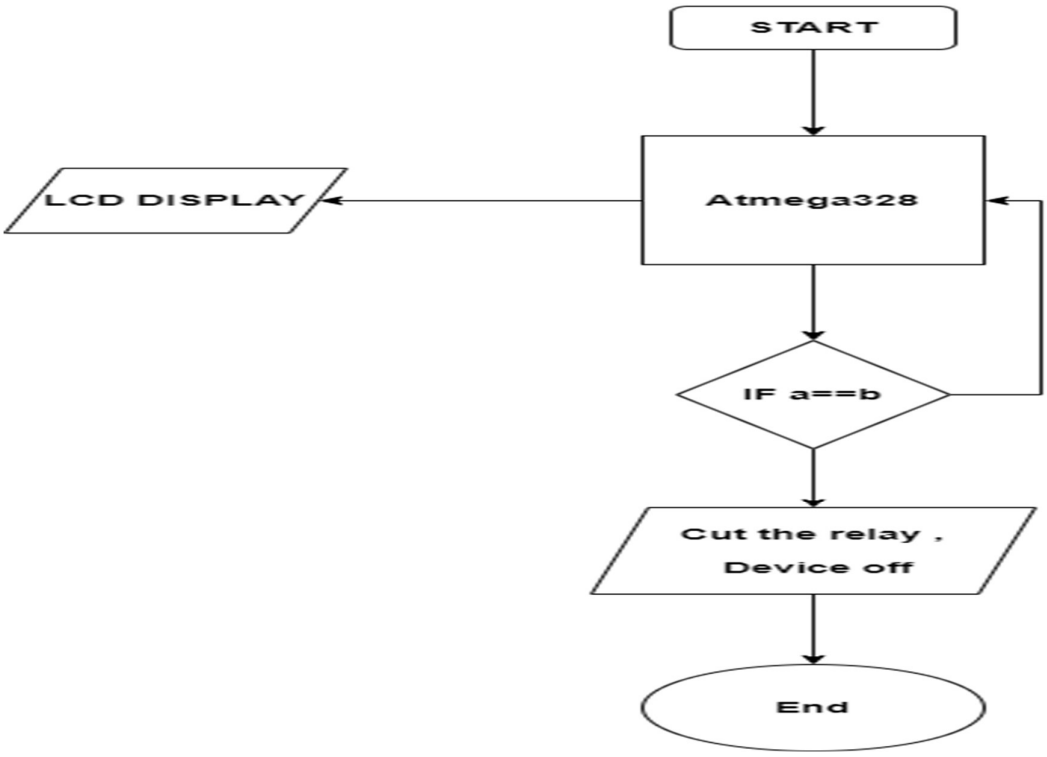


Figure 6 Structure Diagram

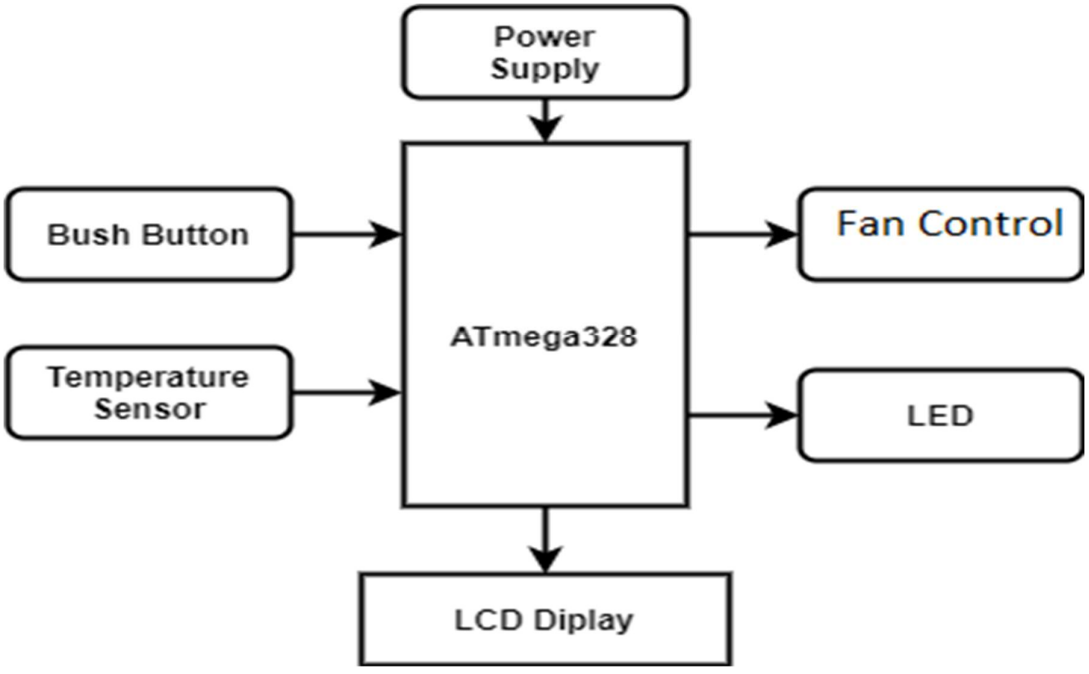


Figure 7 Block Diagram

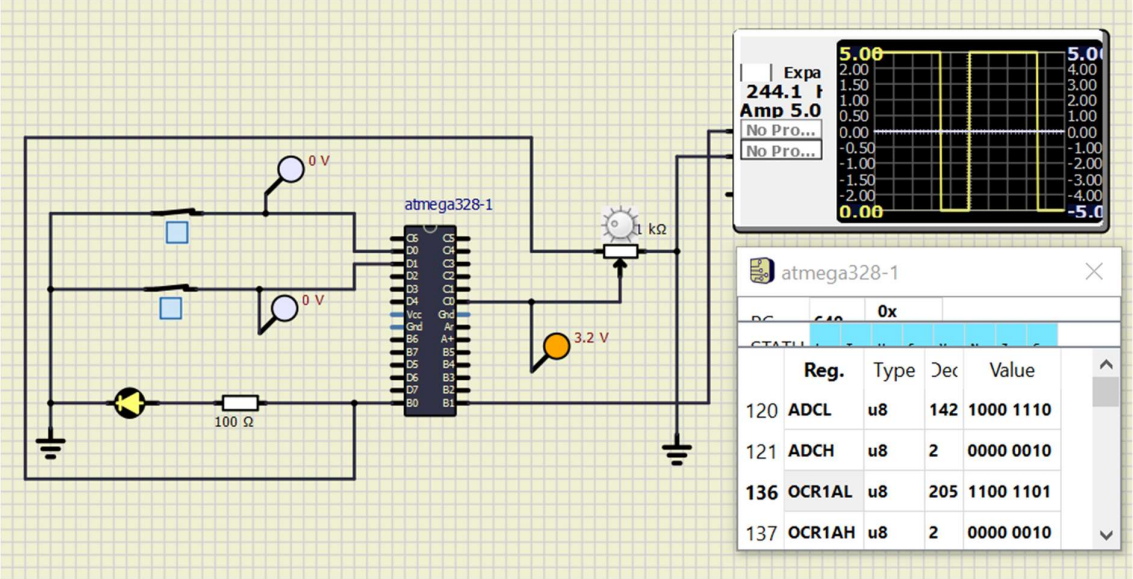


Figure 8 Simulation

Test Plan:

Test ID	Description	Input	Output	Status
01	Is person in room	push button=1	push button=1	PASS
02	Is person in room or not	push button=0	push button=0	PASS
03	Temperature low	Temp=0	Fan=Off	PASS
04	Temperature High	Temp=30	Fan=On	PASS
05	LED ON	Button= && Fan=1	LED=1	PASS
06	LED OFF	Button=0 && Fan=0	LED=0	PASS

Implementation and Summary

Git Link:

Link: https://github.com/pasumarthiharitha/M2_EMBEDDED_HomeAutomation.git

Git Dashboard:

Figure 9 Git Dashboard

Miniproject 3 – Food Court Billing System [Team]

Module: Applied SDLC and Software Testing

The develop a project that helps us effectively manage the Food Court in any institution or outside an institution. This reduces the burden on the owner as the paperwork or calculation work is reduce and other essentials to update.

For simplicity and better understanding of the owner, this software is designed. It would avoid confusion and help operate the software easily. Also, such a software that is easy to use will reduce the work of owner who still maintain all the logs in registers and files. It would be of great benefit as all calculations would be done easily on the click of a button. This reduces the burden on the owner as the paperwork or calculation work is reduce and other essentials to update.

Features:

- For the calculate bills, the user can view their bills after ordering a food.
- For the add orders, the user can add new order of foods.
- For the edit orders, the user can edit their orders information.
- For the display orders, the user can view their orders.
- For the search orders, the user can search their orders.
- For the delete orders, the user can delete their order information.
- For the exit, the user can also exit in the system.

SWOT analysis:

	helpful	harmful
internal	**STRENGTH** ▪ Handling of large number of clients ▪ Multi-lingual staff ▪ High social competences ▪ Reputation in execution / custody	**WEAKNESSES** ▪ Poorly prepared for heterogeneous client requirements ▪ Lack of investment culture and track record ▪ Investment philosophy and investment process of low importance
external	**OPPORTUNITIES** ▪ Taxes are not an issue ▪ Large asset base per client / mandate ▪ Global market with minimal barriers ▪ Proponent timing to enter the market because of low interest rates	**THREATS** ▪ Strong regulatory environment ▪ High fee negotiating power of clients ▪ Strong reliance of clients on external investment consultants ▪ Global competition intense ▪ Supplier rather than partner status

Figure 10 Swot analysis

a) Strength:

This system is a keeping track of billing records, menus and extra food items.

b) Weakness:

All the staff needs to be trained on the software. If there is a power failure, the hotel runs a high risk of losing all the stored information.

c) Opportunity:

This project can be merged with any major projects in future where meals and their monthly calculations need to be done.

d) Threat:

If there is a virus attack the stored information might get corrupt.

Requirements:

4W's and 1 H's:

Who:

It can be used by the owner of the food court to update and to use it freely

What:

A user-friendly application for used to check update in food court daily

When:

As the customers in their recess time use food court inside the company for their food consumption they will need a management system to check today's update

Where:

Used in all mess centres running inside a company for owner's benefit

How:

It can be used in a mobile app easily or can login in a PC.

1]HIGH LEVEL REQUIEMENTS:

ID	Description
HLR1	Customer should be able to add item via item.
HLR2	Customer should be able to search items from menu function.
HLR3	Customer should be able to see their order on display function.
HLR4	Customer should be able to edit their orders.
HLR5	Customer should be able to search item via name or item code.
HLR6	Application should be able to do all calculation that are required to generate bill amount.
HLR7	Customer should be able to delete the item from ordered list.

2]LOW LEVEL REQUIEMENTS:

ID	Description
LLR1	Login Page off Food Court.
LLR2	Enter user and password
LLR3	Newly added details should be display
LLR4	Item name, quantity, rate should be removed
LLR5	Item name, item number and item rate should be there while generating bill
LLR6	Application should return exact final bill.

Flow chart: 1

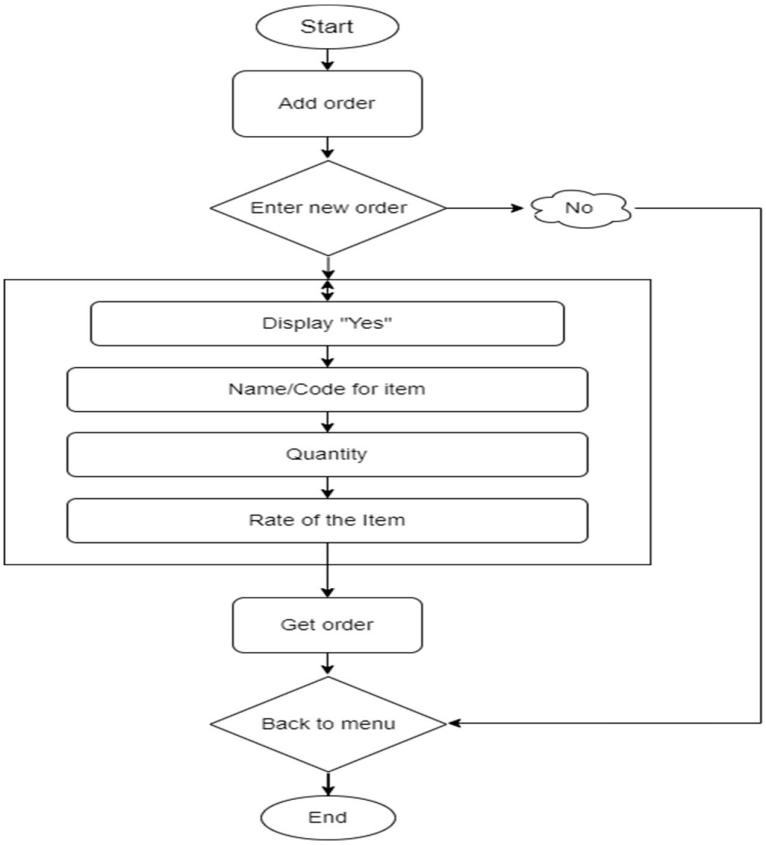


Figure 11 Flow chart 1

Flow chart: 2

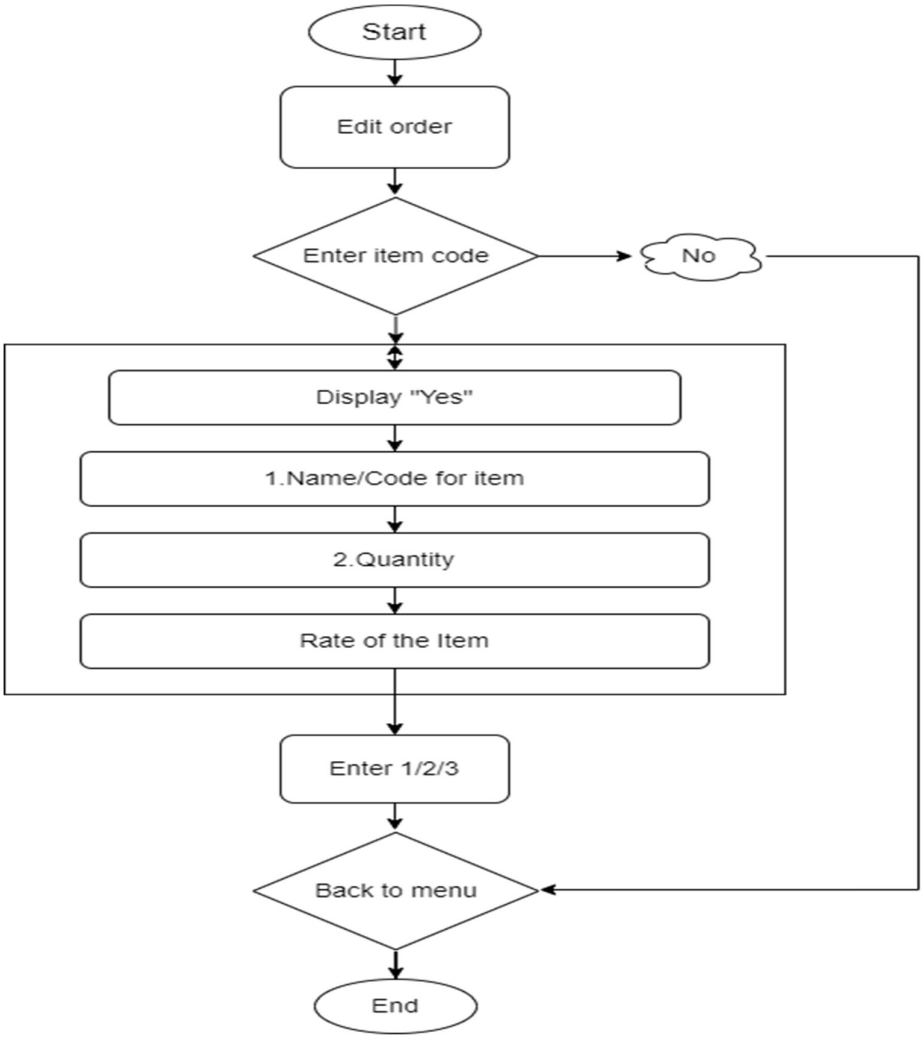


Figure 12 :Behaviour diagram

High level diagram:

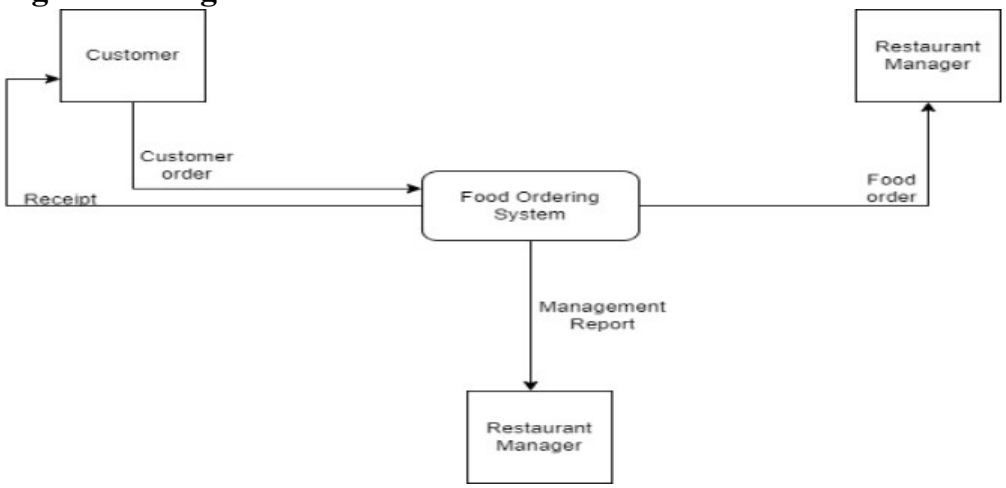


Figure 13High Level Daigram

Low level diagram:

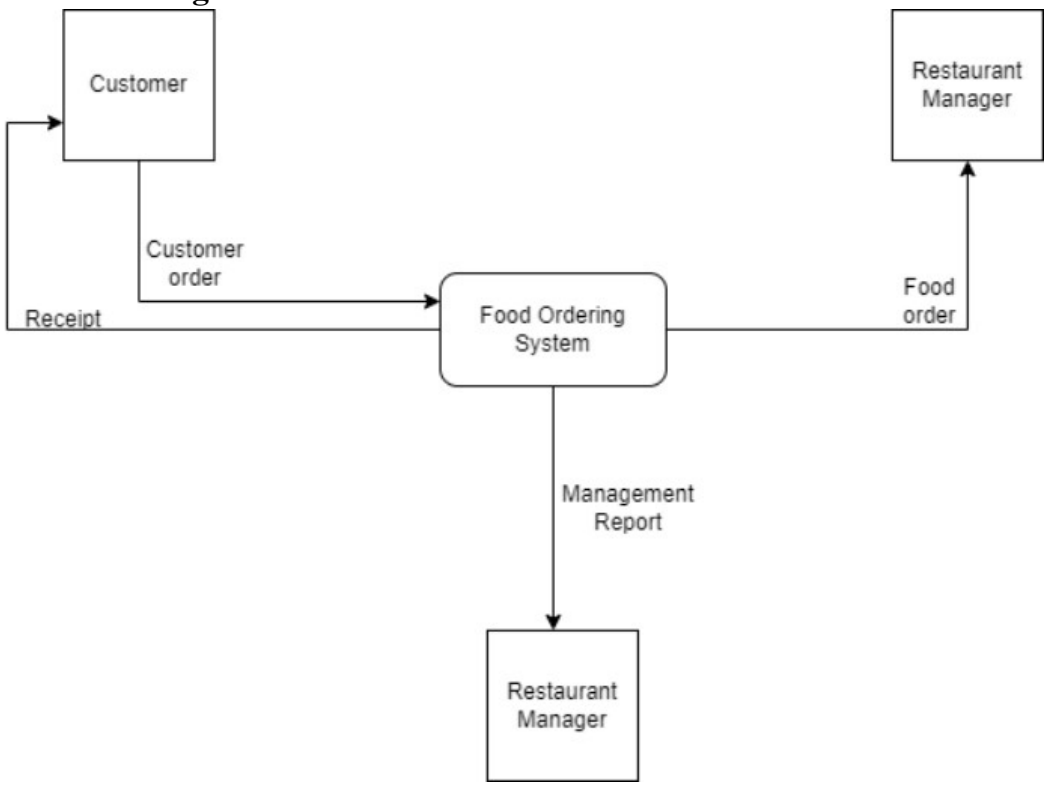


Figure 14 Low Level Diagram

User Flow Diagram:

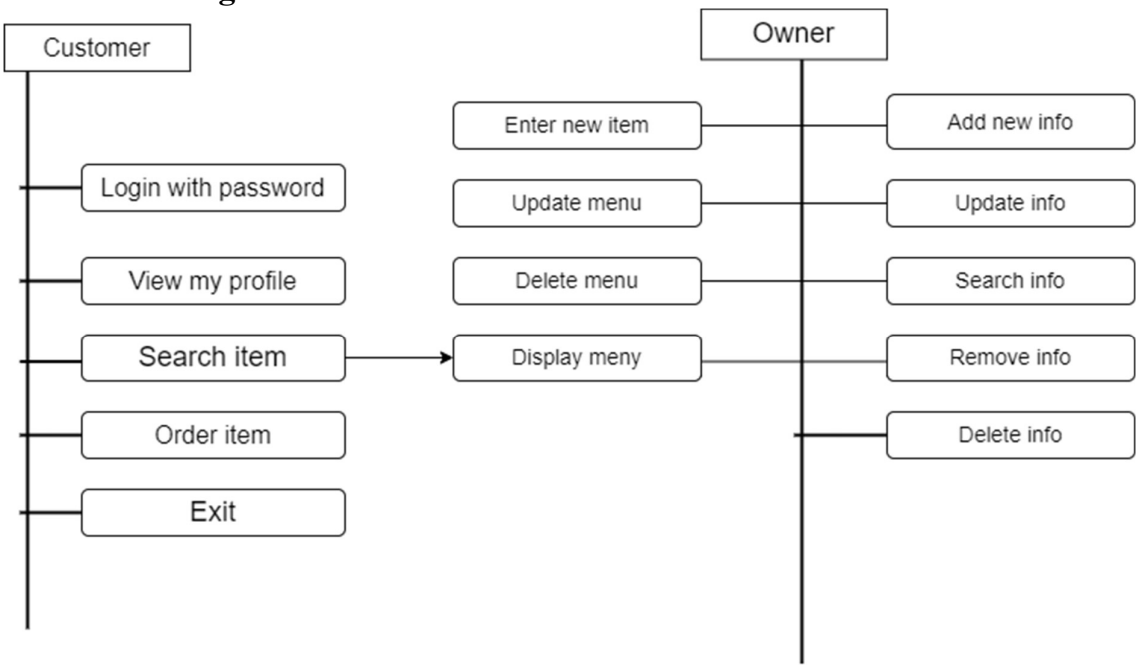


Figure 15 User Flow Diagram

Structure Diagram:

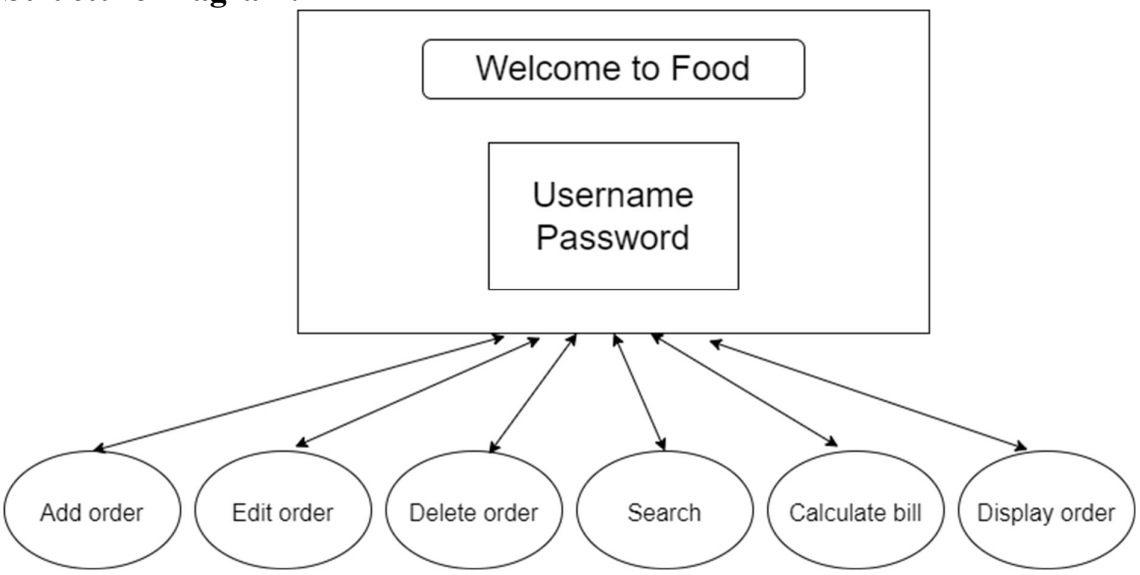


Figure 16 Structure Diagram

Test Plan:

High Level Test Plan:

TEST_ ID	Description	Expected I/P	Expected O/P	Actual O/P	Type Of Test
HLR_1	Login to system	Provide proper Username with character length of 10	entered username	entered username	Requirement Based
HLR_2	Login to system	Provide proper password with character length of 10	Login successful	Login successful	Requirement Based
HLR_3	Providing items that you want to add	User Choice	Added Successfully	Added Successfully	Requirement Based
HLR_4	Display the menu	----	Added Items is Displayed	Added Items is Displayed	Requirement Based
HLR_5	Bill Calculation	Choice	Customer's Bill	Customer's Bill	Requirement Based

Low Level Test Plan:

TEST_ID	Description	Expected I/P	Expected O/P	Actual O/P	Type Of Test
LLR_1	Login to system	username and password are Incorrect (in case numbers or exceed length)	Login is Unsuccessful	Login is Unsuccessful	Requirement Based
LLR_2	Edit Item	Provide Item code	Item Edited	Item Edited	Requirement Based
LLR_3	Delete Item	Provide Item	Item Deleted	Item Deleted	Requirement Based
LLR_4	Search Item	Provide Item Code	Item Searched	Item Searched	Requirement Based
LLR_5	exit operation	----	Exit Successfully	Exit Successfully	Requirement Based

Implementation and Summary:

Git Link:

Link: https://github.com/GENESIS2021Q1/Applied_SDLC-Dec_Team_49.git

Individual Contribution and Highlights:

Summary:

- Add orders
- Edit orders
- Display orders
- Search orders
- Delete orders
- Calculate bill

Individual Contribution:

1. Creating folders
2. Suggestion of ideas to team
3. In coding Display order and items are done
4. Issues raised and rectified the other issues

Mini project 4 – Calendar Automation [Team]

Modules: - OOPS with Python

Requirements:

High Level Requirements:

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

Link for template standard input template:
https://docs.google.com/spreadsheets/d/1EWYp_1iyK2wLMfKGJOiTJAk5WexZusCP/edit?usp=sharing&ouid=113003694561146884677&rtpof=true&sd=true

- Using the template above, training schedule can be added month wise and initiatives wise
- The name of the input excel sheet MUST be named as "Test vector"(as shown in template)
- Along with the Test vector sheet, "Key" sheet MUST be present under the columns assigned as in the template
- The "Key" sheet must contain all times the 6 fixed initiatives with their respective codes and total list of course code and course title to refer for corrections while writing to output files
- Appending additional slots for existing courses is possible by adding just the additional slots in the input file for the same course

Requirements for updating Master calendar using Master calendar as input:

Link for template:
2 Slots format - M/A:
https://docs.google.com/spreadsheets/d/1jtKnXV12VE1fH20CGDo4B3uNWRTAhQCWz-hHUDWUe3I/edit?usp=sharing

4 Slots format - M1/M2/A1/A2:
https://docs.google.com/spreadsheets/d/1jVheSPZkOtfNKRNoc_858nwk2UaHCe0gExTNZfZ8vxA/edit?usp=sharing

- Any of the two templates can be used for updating Master calendar month wise on to the drive
- The blocked slots must have the corresponding initiative code in the cell according to the key as shown in the sample data in the template
- The name of the sheet must be the name of the month to be updated
- The "Key" sheet must be present with the fixed list of initiatives and initiative code

App deployment

- The app is deployed on Heroku servers.
- To add/modify new features, you will be required to install HEROKU CLI link

- After installation, open terminal in working directory and enter the following commands:
  - "heroku git:clone -a gea calendar"
  - login using heroku credentials
- After pulling and making changes, enter the following commands to push app and deploy on server
  - Git adds.
  - git commit -m "commit message"
  - git push heroku master

Additional features for V1 to do:

- Update key sheet by appending new initiatives/courses list
- Check for duplicate course entries in input file
- Using built in libraries to identify number of days in month, current year and highlight weekend and holidays
- Function to remove a course schedule
- Read multiple months data in one sheet as input file (currently takes data one by one month)
- Calculate individual faculty load

Git Link:

https://github.com/Pradnya579/GENESIS2021-OOP-Python_Team_46.git

Individual Contribution and Highlights:

1. Improved implementation of Python Programming
2. Source code management using Git Hub

Role in Project Team:

Done Programming for calendar Automation

Miniproject 5 – Hyundai Project [Team]

Module: Applied Model Based Design Module

Contributors:

S. No	NAME	PS Number	Feature
1	Y. Lakshman Swami	40021033	Door locking
2	P. Haritha	40021034	Sunroof control
3	V. V K Mallikarjun	40021038	Sunroof control
4	V. Sai Kumar	40021040	Window control
5	S. Usha Rani	40021045	Power Window
6	N. Kesava Kumar	40021058	Security Systems
7	T. Lakshmi Narayana	40021060	Wiper Control

Requirements:

Door Locking System:

Power door locks (also known as electric door locks or central locking) allow the driver or front passenger to simultaneously lock or unlock all the doors of an automobile or truck, by pressing a button or flipping a switch. Nearly every car model today offers this feature as at least optional equipment.

Sunroof control:

A sunroof is a movable panel that opens to uncover a window in an automobile roof, allowing light and fresh air to enter the passenger compartment. Sunroofs can be manually operated, or motor driven, and are available in many shapes, sizes and styles. While the term sunroof is now used generically to describe any glass panel in the roof, the term "moonroof" was historically used to describe stationary glass panes rigidly mounted in the roof panel over the passenger compartment.

Power Windows:

Power windows or electric windows are automobile windows which can be raised and lowered by pressing a button or switch, as opposed to using a crank handle. Power windows are usually inoperable when the car is not running. This is primarily a security feature. It would be a simple thing to allow electric power windows to be operable when the ignition is turned off, however it would also make the car much easier to steal. Some systems offer the compromise of leaving power applied to the windows until a passenger door is opened at which time the window power is removed.

Security System:

A car alarm is an electronic device installed in a vehicle to discourage theft of the vehicle itself, its contents, or both. Car alarms work by emitting high-volume sound (often a vehicle-mounted siren, klaxon, pre-recorded verbal warning, the vehicle's own horn, or a combination of these) when the conditions necessary for triggering it are met. Such alarms may also cause the vehicle's headlights to flash, may notify the car's owner of the incident via a paging system, and may interrupt one or more electrical circuits necessary for the car to start. Although inexpensive to acquire and install, the effectiveness of such devices in deterring vehicle

burglary or theft when their only effect is to emit sound appears to be negligible. He individual triggers for a car alarm vary widely, depending on the make and model of the vehicle, and the brand and model of the alarm itself (for aftermarket alarms). Since aftermarket alarms are designed to be universal. Although car alarms of some kind have been available since the beginning of the automobile era, the dramatic increase in their installation in the 1980s and 1990s coupled with the fact that nearly all types of car alarms are easily triggered accidentally (frequently because of high sensitivity settings) means that people who hear them often ignore them.

Wiper Control:

A windscreen wiper, windshield wiper or wiper blade (American English) is a device used to remove rain, snow, ice, washer fluid, water, and/or debris from a vehicle's front window so the vehicle's operator can better see what's ahead of them. Almost all motor vehicles, including cars, trucks, buses, train locomotives, and watercraft with a cabin—and some aircraft—are equipped with one or more such wipers, which are usually a legal requirement. On some vehicles, a windscreen washer system is also used to improve and expand the function of the wiper(s) to dry or icy conditions. This system sprays water, or an antifreeze window washer fluid, at the windscreen using several well-positioned nozzles. Most wipers are of the pivot (or radial) type: they are attached to a single arm, which in turn is attached to the motor. These are commonly found on many cars, trucks, trains, boats, airplanes, etc. Wipers may be powered by a variety of means, although most in use today are powered by an electric motor through a series of mechanical components, typically two 4-bar linkages in series or parallel.

References:

https://mechvibesblog.com/control-modules/

https://youtu.be/7zzpTH9Hl-s

https://youtu.be/4KS8jUCCbkQ

https://youtu.be/g6MgZY6Gbc8

https://youtu.be/PdH792tFV7M

https://youtu.be/r6gbQTt1Blc

Design:

This project was implemented using MATLAB.

Individual contribution:

Design the MATLAB scripting for Placement Data Full Class data sheet from google information.

Mini project 6 – Wiper Control [Team]

Module: - Mastering Microcontrollers with Embedded Driver Development Module

WIPER CONTROL SYSTEM

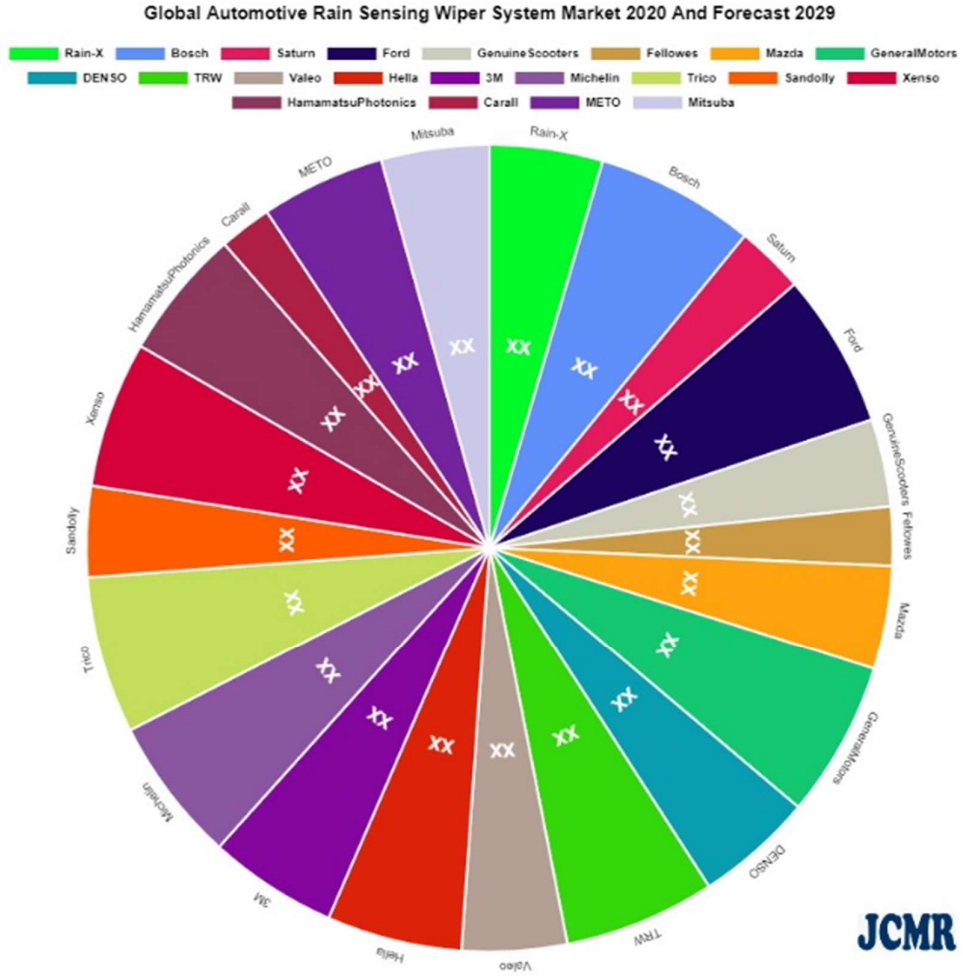
Introduction:

Automotive wipers form an essential part for any vehicle. They perform to remove water, ice, snow, and dust from a windshield of a vehicle. An automotive wiper is either powered by an electric motor or pneumatic power. Almost all motor vehicles including cars, trucks, buses, train locomotives and watercraft with a cabin are equipped with one or more such wipers. The automotive wiper market is multiplying as there is an exponentially increased production of automobiles globally.

Features:

- To achieve high safety
- To reduce manpower
- To increase the efficiency of the vehicle
- To reduce the workload
- To reduce the vehicle accident
- To reduce the fatigue of workers
- To high responsibility
- Less Maintenance cost

State of Art



SWOT Analysis:

Strength:

- It is possible to operate Manually/automatically by proving On/Off switch
- Improve Visibility of car in rain.Can drive easily in any climatic situation.

Weakness:

- This system applied in the case of water falling on the class only.
- Addition cost is required to install this system to four-wheeler.

Opportunities

- To increase automation in vehicle driving system
- To dispense with troublesome wiper operation needed when rainfall condition change or driving condition change, including the car speed and entry to or exit from tunnels.
- To operate the wiper with response to changing rainfall or driving conditions, thus keeping the driver's windshield clear.

Threats

- Dust particles and non-conductive particles accumulated on the surface of sensors cannot be detected by conductive sensors.

4W's & 1H:

Who:

- A wiper speed control system for an automotive wiper controls the operational speed of a wiper in accordance with rain conditions.

What:

- Vehicles are now available with driver-programmable intelligent (automatic) windscreen wipers that detect the presence and amount of rain using a rain sensor.

Where:

- It is located underneath the dashboard, above the brake and accelerator pedal, and is responsible for the complete operation of the windshield wiper system.

When:

- Whenever the water hit a dedicated sensor that located on windscreen, it will send a signal to move on the wiper motor. Once water is not detected by sensor, the wiper will automatically stop. This will help the driver to give more concentration and reduce the car accident probability.

How:

Windshield wipers are controlled by the stalk on the right side of your steering wheel. Simply moving the stalk down will turn your windshield wipers on. Moving the stalk down will turn your wipers on.

BLOCK DIAGRAM:

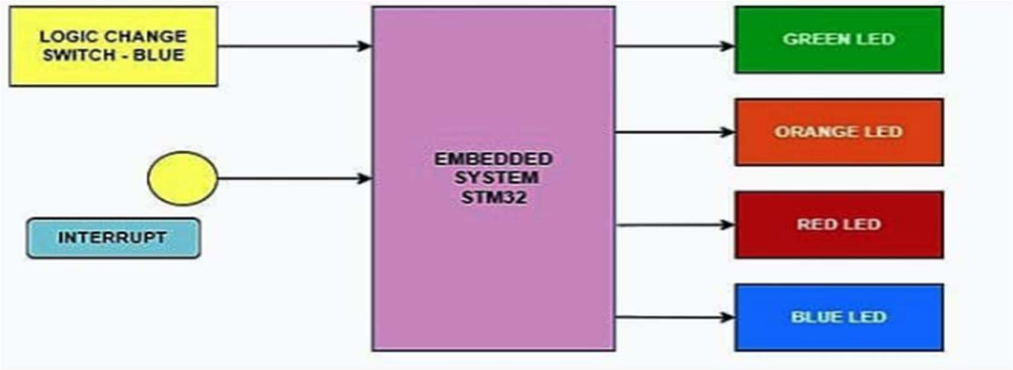


Figure 17 Block diagram

FLOW CHART:

Figure 18 Flow chart

CASE STUDY:

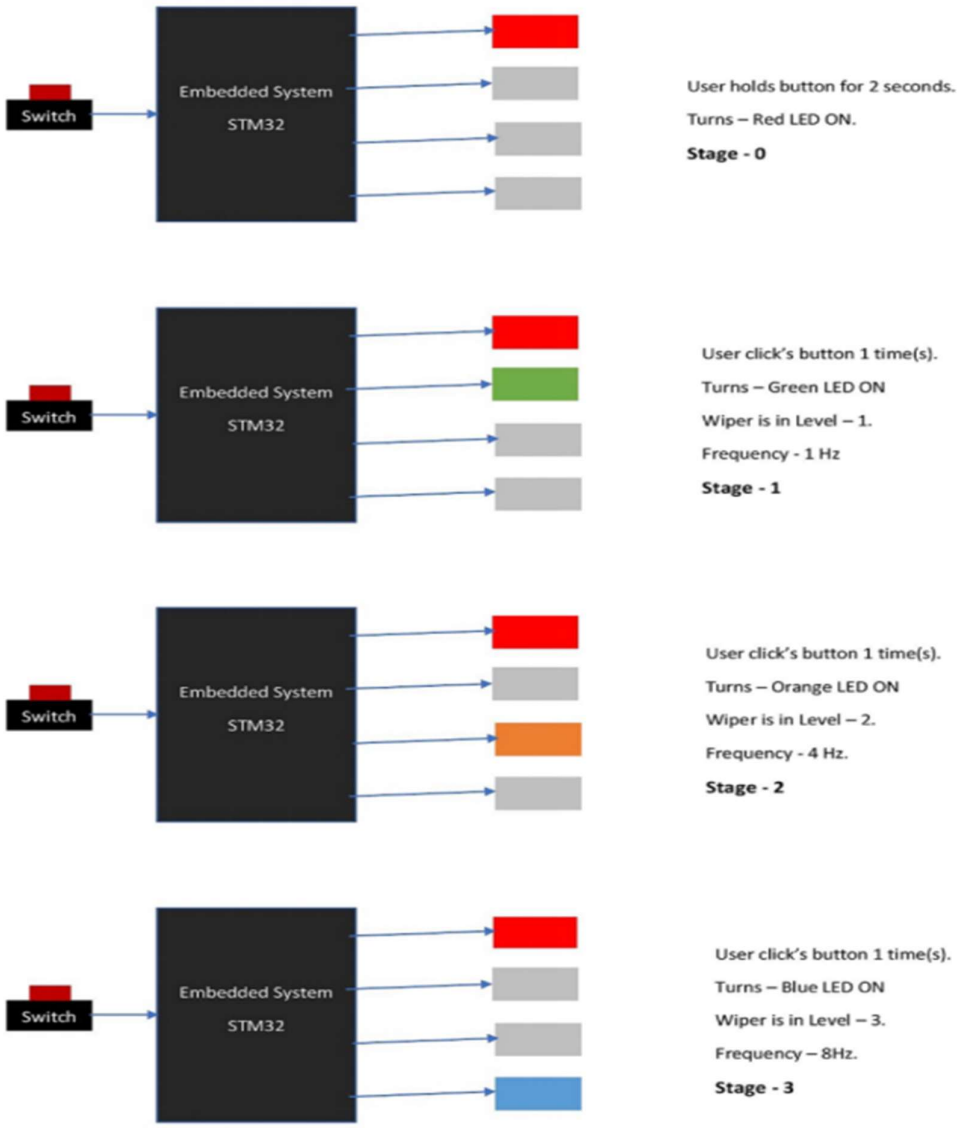
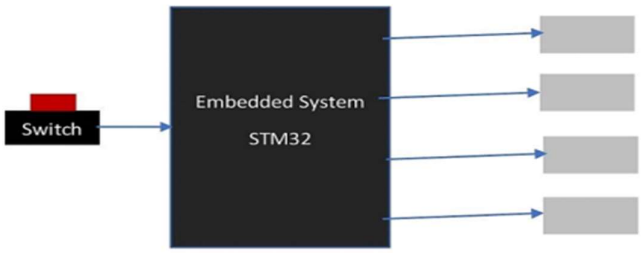


Figure 19 Case Study



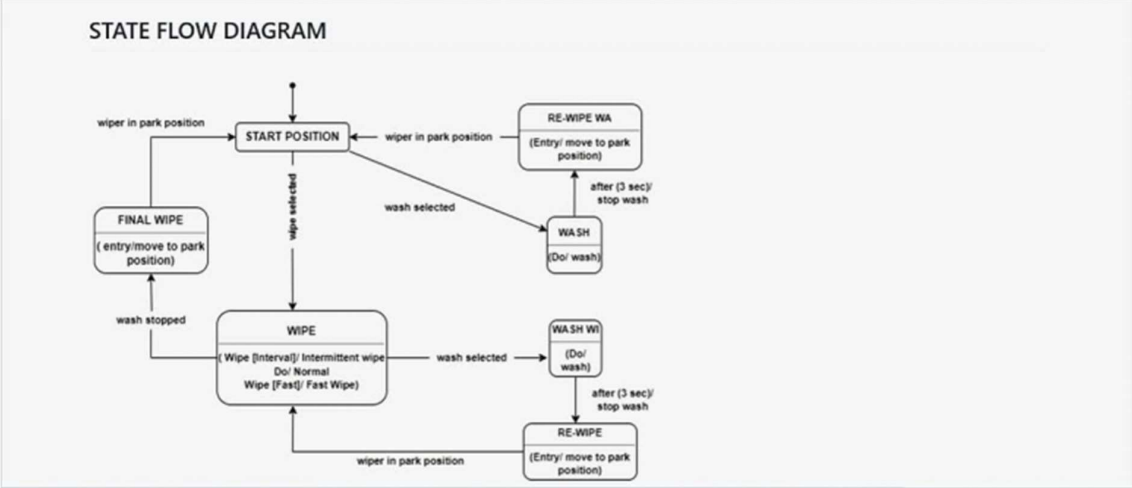
User holds button for 2 seconds.

Turns – All LED OFF.

Stage - 0

According to the requirement user can change the wiper levels, this process is shown in forward direction, and the three LEDs other than red LED can be changed from level to level.

STATE FLOW:



SYSTEM DESIGNS:

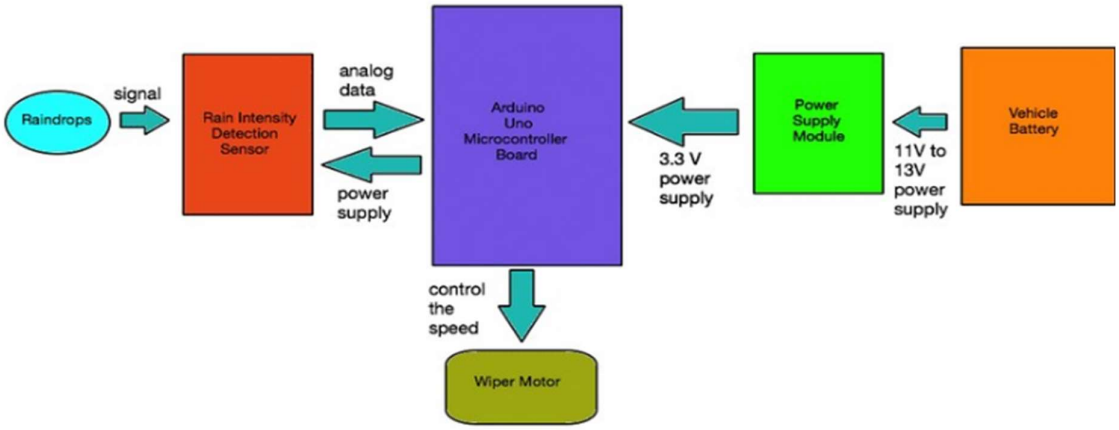
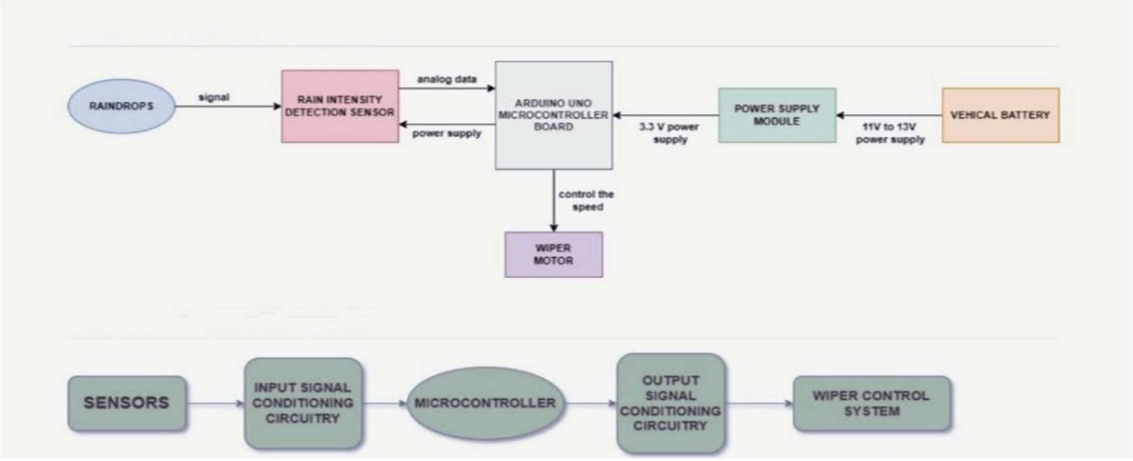


Figure 20 System Designs



High Level Requirements:

ID	Description
HLR1	These systems detect droplets of rain on the windshield and automatically turn on and adjust the wiper system in accordance with the level of precipitation.
HLR2	A windscreen wiper or windshield wiper is a device used to remove rain, snow, ice and debris from a windscreen or windshield.
HLR3	Quality and reliability wiper systems meet the highest technical requirements and are the basis for vehicles with sophisticated features.
HLR5	Almost all motor vehicle, including trains, aircraft, and watercraft, are equipped with such wipers, which are usually an essential requirement.
HLR6	Our project brings forward this system to automate the wiper system having no need for manual intervention.

Low Level Requirements:

ID	Description
LLR1	A new mechatronic reversing system can now be used to clean the windshield with two wiper arms, whereby one wiper arm is powered directly and the other indirectly using a connection link.
LLR2	Wiper motor is automatically ON during the time of rainfall.
LLR3	Existing system manually used control stalk to activate wiper and the process of pulling up wiper is difficult to be handled.
LLR4	Lower-level parsing. Under the hood, the Requirement class does most of the heavy lifting. class requirements.
LLR5	These systems detect droplets of rain on the windshield and automatically turn on and adjust the wiper system.

Implementation and Summary:

Git Link: *https://github.com/GENESIS-2022/MasteringMCU-Team36.git*

Individual Contribution and Highlights:

1. Wiper System using C Programming
2. Source code management using GitHub
3. REQUIREMENTS & Implementation (start-up's, STM32F407XX.H)

Role in Project Team:

1. Programmer: Done Programming for Wiper System
2. Integrator: Integrated all the codes
3. Tester: Writing Test cases and testing the integrated code

Miniproject 7 – LAMBHORGINI[Team]

Module: Automotive Systems

Requirements:

In this project Lamborghini we have taken following features and I have contributed to Parking System Feature

1. Wiper Control System
2. Sunroof control system
3. Power window system
4. Seat belt control

Research:

Automation is a necessity in our day-to-day life because it not only seeks to improve the quality of life for humans at both home and work.

It allows the distribution of both quality products and services to be made available at faster rates, reduces time and human error.

These days the technology is developing at very high-speed innovation and advancements is there in all the sectors.

SWOT- Strengths, and Weakness, Opportunities Threats:

Strengths:

Comfortable, Easy to use, Fast and accuracy operation, Easy transportation

Weakness:

High cost

Opportunities:

it's a modern way of approach, High use in modern days

Threat:

if sensor is damages, operation of the system will stop

4W's and 1'H:

WHO:

Any user who is travelling in the car can use this system.

WHAT:

It’s a Body control model of a car which describes different features

WHEN:

There is a requirement of vehicle these systems are used

WHERE:

Used in automatic cars

HOW:

By developing an embedded system which is user friendly and can be implemented without difficulty.

Detail requirements:

High Level Requirements:

Number	Requirements	Description	Features
HLR1	Rain and Humidity sensor	checks the intensity of rain and humidity	Wiper Control
HLR2	Semi-Automatic control	Operated automatically or manually	Wiper Control
HLR3	Speed control	High, Medium, Low	Wiper Control
HLR4	Dry mode	front and back wiper system	Wiper Control
HLR5	Obstacle Sensor	when obstacle detected the operation is delayed for 10seconds.	Power window
HLR6	Both driver and passenger control	operated with the first received command (either by driver or passenger)	Power window
HLR7	Dust detected sensor	window will be closed when the dust is detected	Power window
HLR8	Temperature and Rain sensor	Which detects intensity of rain and temperature	Sunroof Control
HLR9	Obstacle sensors	when obstacle is detected operation of window will be stopped	Sunroof Control
HLR10	Pressure control alarm	alarm buzzes when seat belt is not buckled up even the pressure is applied	Seatbelt Control

Number	Requirements	Description	Features
HLR11	Open airbag	depending upon the tension airbag will be opened	Seatbelt Control
HLR12	buckles and its types	lap belts, diagonal belts, the harness, combination of lap and diagonal belts.... etc.	Seatbelt Control
HLR13	Height adjustment of seat belt	Height adjustment of seat belt	Seatbelt Control

Low Level Requirements:

Number	Requirements	Description	Features
LLR1	Length of wipers	Depends on the model of the car	Wiper Control
LLR2	Direction of wipers	Same and Opposite Directions (180 degrees)	Wiper Control
LLR3	Semi controlled	manually/automatic	Power window and Sunroof
LLR4	Length of the belt	Depends on the model of the car	Seat Belt control
LLR5	Engine On	Can be operated manually or automatically	wiper control, power window, sunroof control
LLR6	Engine Off	Can be operated manually	Power window and Sunroof control

Design:

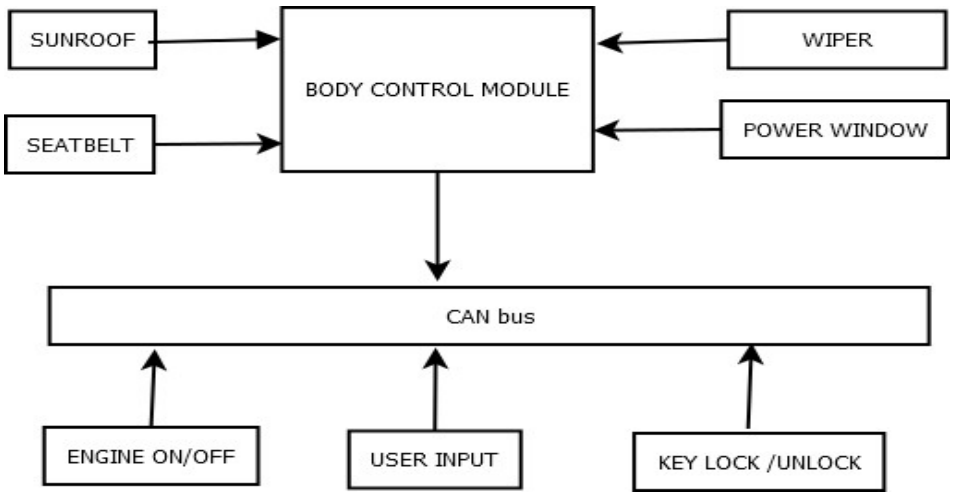


Figure 21 Structure Diagram

Implementation and Summary:

Git Link:

Link: *https://github.com/pasumarthiharitha/Automotive_Lamborghini.git*

Individual Contribution and Highlights:

1. Sunroof Case Study
2. Source code management using GitHub

Role in Project Team:

1. Designer: Done Designing for Project
2. Researcher: Done case study for Sunroof Control

Miniproject 8 – EV Car [Team]

Module: Applied Control Systems and Vehicle Dynamics

Requirements:

1.Battery:

1.Battery type used in this EV is Lithium-ion Polymer.

2.Lithium is also the lightest of all metals. However, lithium-ion (Li-ion) batteries contain no lithium metal, they contain ions. For those wondering what an ion is, an ion is an atom or molecule with an electric charge caused by the loss or gain of one or more electrons.

Battery Features:

3.Lithium-ion batteries are one of the most popular forms of energy storage in the world, accounting for 85.6% of deployed energy storage systems.

2.Motor:

1.BLDC motors are used, which have traction characteristics like high starting torque, high efficiency around 95 98%, etc.

2.BLDC motors are suitable for high power density design approach. The BLDC motors are the most preferred motors for the electric vehicle application due to its traction characteristics.

Motor Features:

3.Electronically commutated

4.High-energy, rare-earth magnets used for rotor field

5.Requires speed control with 6-lead connection (3 power,2 Hall Effects, 1 drain)

6.Rated speed 2500 RPM; minimum 150-200 RPM

7.Linear speed torque curves

8.Built-in tach pulse for economical speed readout

9.Encoder options for servo performance.

3.Controller:

1.Programmable BLDC motor controller provides efficient smooth and quite control for electric vehicles which operate 36V/48V battery system.

controller Features:

1.E- Braking will release the motor from the controller if brake is applied.

2.Over current protection to protect controller during faulty conditions or short circuit.

3.Low voltage detection ensures Battery life.

4.Pedal assist mode controls the motor speed based on the speed of peddling.

5.Accelerator fault protection to prevent runway.

6.Provision for 120°/ 60° Selection.

7.Speedometer output.

8.Brake inputs with high active and low active provision.

9.Speed limit control provision.

4.Inverter:

1.The traction inverter converts energy from the vehicle's battery in order to drive the motors in the drivetrain. This key component has a direct impact on road performance, driving range and reliability of the vehicle also because of their weight and size.

2.Subject to all the possible stress found in a road vehicle from heat and vibrations, these converters must be able to handle high power and currents along with the associated Electro Magnetic Compatibility (EMC) challenges as well as provide fail-safe operation to ensure dependability and safety for the driver and passengers.

3.To help developers increase the automotive inverter's power efficiency and reduce size and weight, ST has a wide offer of discrete semiconductors including AEC-Q101 qualified IGBTs and both silicon and silicon-carbide (Sic) MOSFETs and diodes, AEC-Q100 qualified galvanically isolated IGBT and MOSFET gate drivers and SPC5 32-bit automotive microcontrollers for designing scalable, cost-effective, and energy-efficient EV traction inverter solutions.

Inverter features:

4.Traction inverters are typically capable of transferring power in the 20 to 100 kW range, with switching voltages in the 200 V to 800 V range and currents in the hundreds of amperes.

Implementation and Summary:

Submission: Submitted in GEA Learn

Individual Contribution and Highlights

1. Done in MATLAB Script

Role in Project Team:

1. Done MATLAB scripting for EV Car
2. Researcher: Done case study for EV Car

Mini project 9 – SUNROOF CONTROL [Individual]

Module: Classic AUTOSAR Basic to Intermediate

Requirements:

High Level Requirement:

High level Requirement	Description
HLR_1	Light will be on when Door is open
HLR_2	Light will be in on state until all the door is correctly closed
HLR_3	Dashboard Light will ON when the car is unlocked
HLR_4	Lights will be OFF after 10 sec when the lock button pressed on the key
HLR_5	In Night Footstep light is automatically ON
HLR_6	Lights can be turn off and on manually with the switch

Low Level Requirement:

Low level Requirement	Description
LLR_1	Voice command to turn on and off the light
LLR_2	Lights will be turned on when unlocks the car with the remote key from outside
LLR_3	Add Multicolour LED
LLR_4	The light will turn-off after 5 seconds when we lock the car through the key
LLR_5	Safety Indicator

Design:

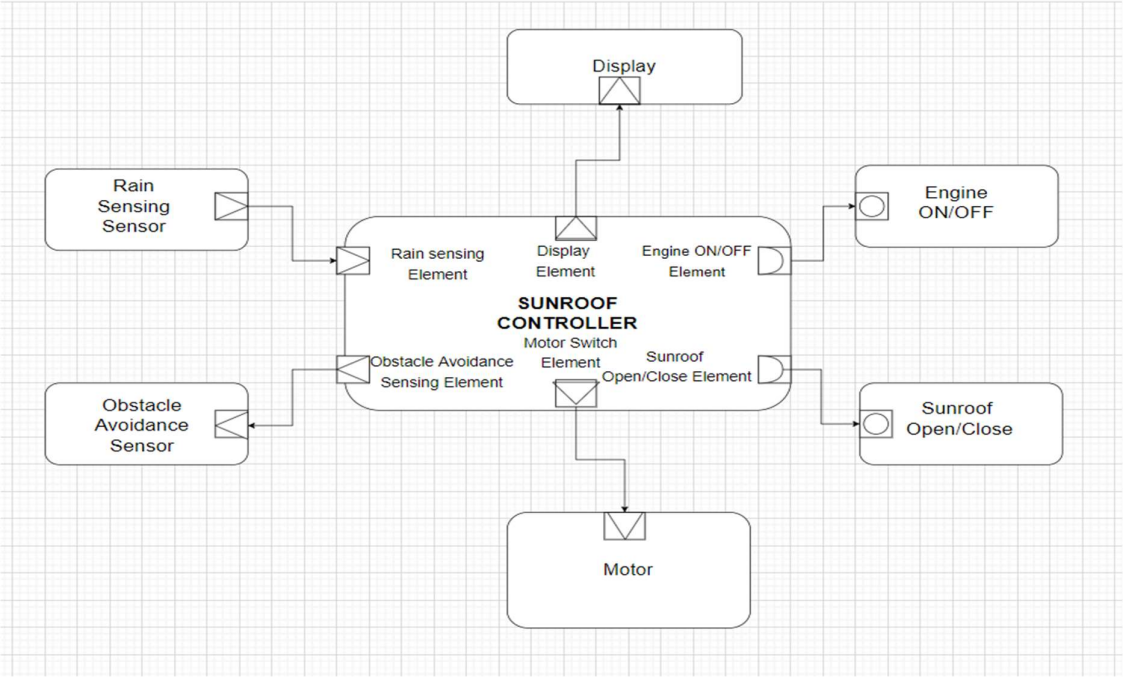


Figure 22 VFB Diagram

Implementation and Summary:

Git Link:
Link: *https://github.com/pasumarthiharitha/Automotive_Lamborghini.git*

Individual Contribution and Highlights:

1. Sunroof control Case Study
2. Source code management using GitHub
3. Atomic Component
4. SWC Internal Behaviour
5. SWC Implementation