Ticket Portal System

Team Members

1. Abhishek Kumar M (Team Leader)
2. Arnold Binu Varghese
3. Emi Mary Rose
4. Shwetanjali
5. Siranjeev M

Git Hub Link: https://github.com/ABHISHEK-KUMAR-M/Capestone-project

CONTENTS

Problem Statement

- Employee issues handled using manual and unstructured methods
- No centralized ticket tracking system
- Lack of proper prioritization and SLA enforcement
- No visibility into ticket status and ownership
- Inefficient communication between employees and support teams

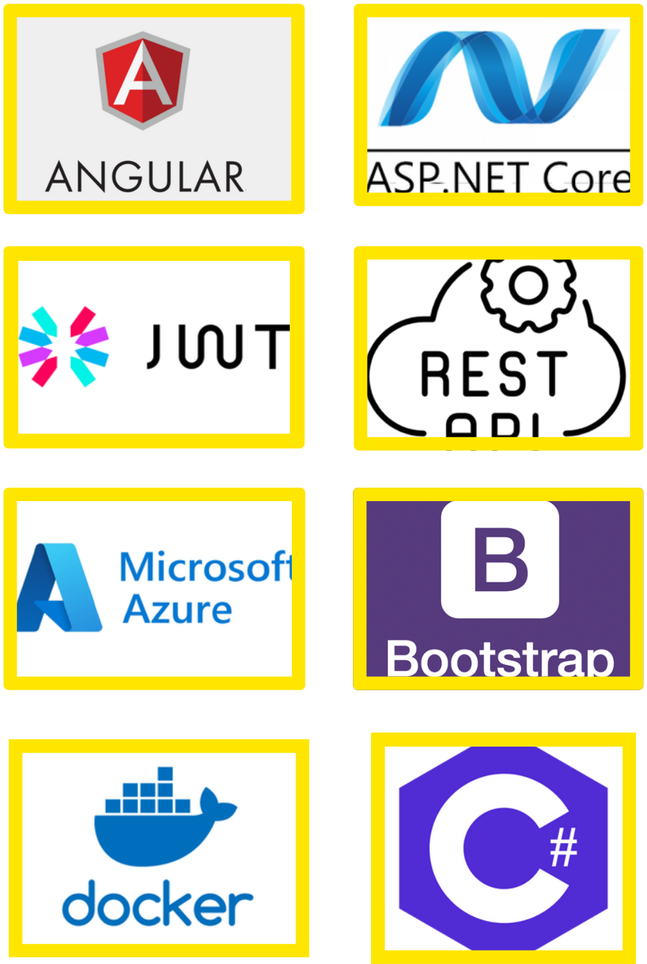
Project Objectives

- Build a centralized ticket management system.
- Ensure role-based access for Admin and User (Employee).
- Enable real-time communication through ticket replies.
- Provide secure authentication using JWT.

About the Project

- Web-based application for managing employee support tickets.
- Allows users to:
  - Raise tickets.
  - Assign tickets.
  - Supports ticket reply conversations.

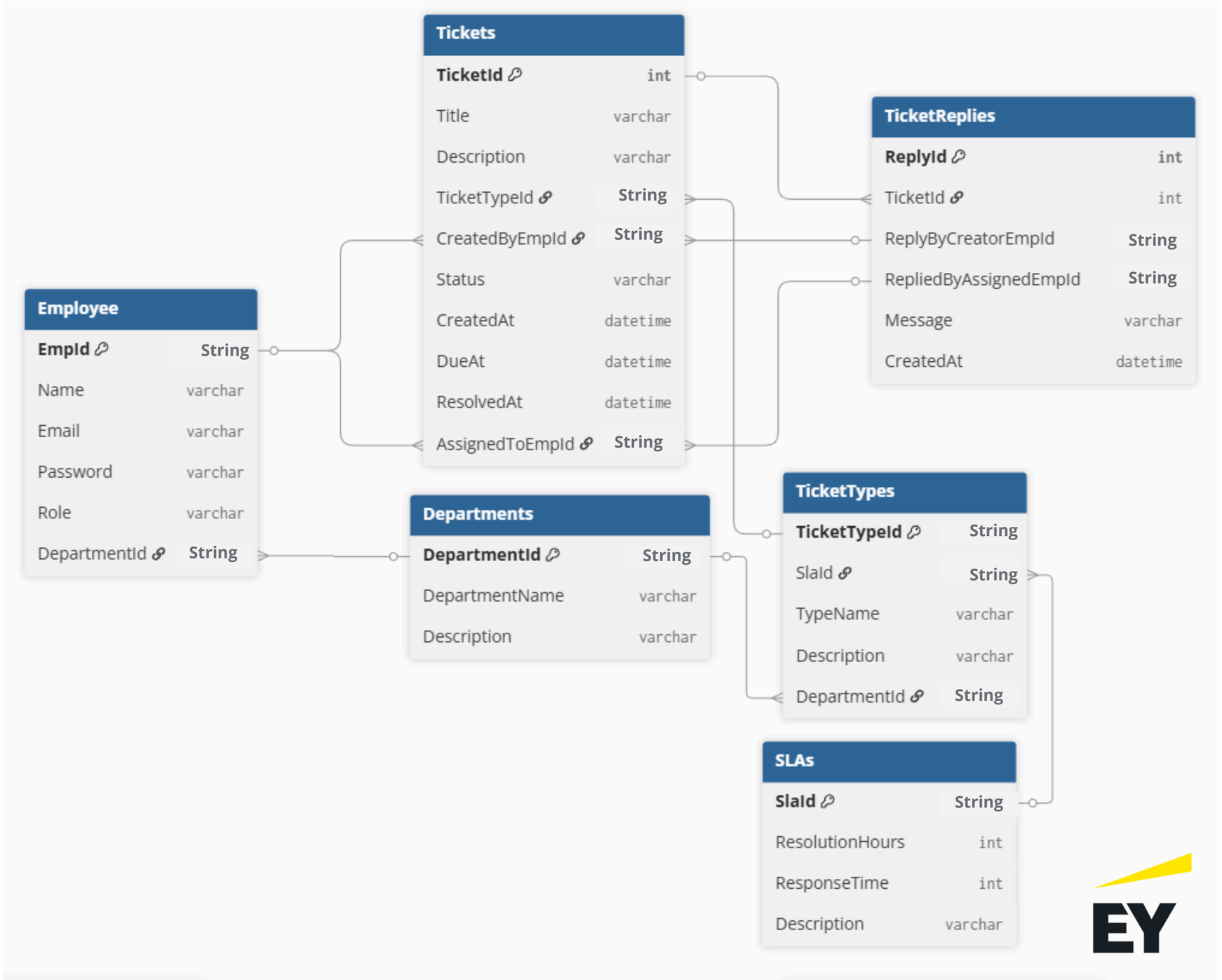
Tech Stacks Used



- Frontend:
  - Angular
- Backend:
  - ASP.NET Core Web API (Restful APIs)
  - Fluent API (Entity Framework Core Configuration)
  - C#
- Authentication & Security:
  - JWT Token-Based Authentication & Authorization
- Database:
  - Azure SQL Server
- ORM / Data Access:
  - Entity Framework Core (Code-First Approach)
- Deployment & Cloud:
  - Docker (Containerization)
  - Azure Blob Storage (Frontend Hosting)

01

ER DIAGRAM



Relationships:

- **One Department → Many Employees**
- **One Department → Many Ticket Types**
- **One SLA → Many Ticket Types**
- **One Ticket Type → Many Tickets**
- **One Employee → Many Tickets (Created / Assigned)**
- **One Ticket → Many Ticket Replies**

ER Diagram

01

Employee

- Stores system user details
- Attributes: Employee ID, Name, Email, Password, Role
- Each employee belongs to one department
- Employees can create tickets, get assigned tickets, and post replies

02

Ticket Type

- Categorizes tickets based on issue type
- Attributes: Ticket Type ID, Type Name, Description
- Each ticket type belongs to one department
- Each ticket type is associated with one SLA

03

Department

- Represents organizational departments
- Attributes: Department ID, Name, Description
- One department has multiple employees
- One department manages multiple ticket types

04

Ticket

- Represents support issues raised by employees
- Attributes: Ticket ID, Title, Description, Status, Created At, Due At
- Created by one employee and assigned to one employee
- Each ticket belongs to one ticket type
- One ticket can have multiple replies

05

SLA

- Defines response and resolution time limits
- Attributes: SLA ID, Response Time, Resolution Hours, Description
- One SLA can be linked to multiple ticket types
- Used to identify overdue tickets

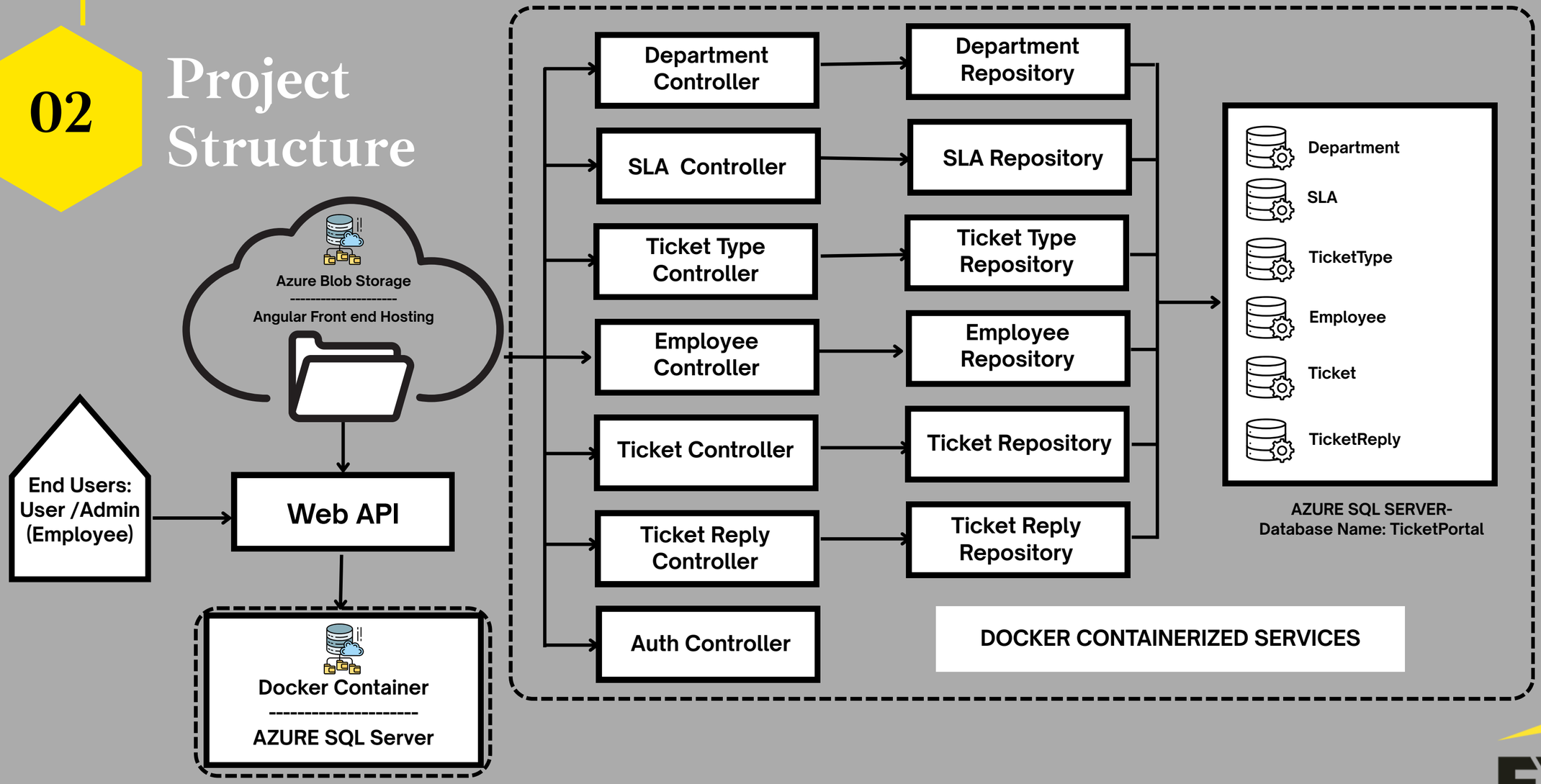
06

Ticket Reply

- Stores conversation messages for tickets
- Attributes: Reply ID, Message, Created At
- Each reply belongs to one ticket
- Replies can be posted by ticket creator or assignee

02

Project Structure



01. Department

Method	Functionality
CreateDepartmentAsync()	Creates a new department
UpdateDepartmentAsync()	Updates department information
DeleteDepartmentAsync()	Deletes department if no dependenc
GetDepartmentByIdAsync()	Fetches department by ID
GetAllDepartmentsAsync()	Retrieves all departments

02. Employee

Method	Functionality
CreateEmployeeAsync()	Registers a new employee
UpdateEmployeeAsync()	Updates employee details
DeleteEmployeeAsync()	Deletes employee after dependency validation
GetEmployeeByIdAsync()	Fetches employee by employee ID
GetAllEmployeesAsync()	Retrieves all employees
GetByDepartmentIdAsync()	Retrieves employees belonging to a department
LoginEmployee()	Authenticates employee credentials

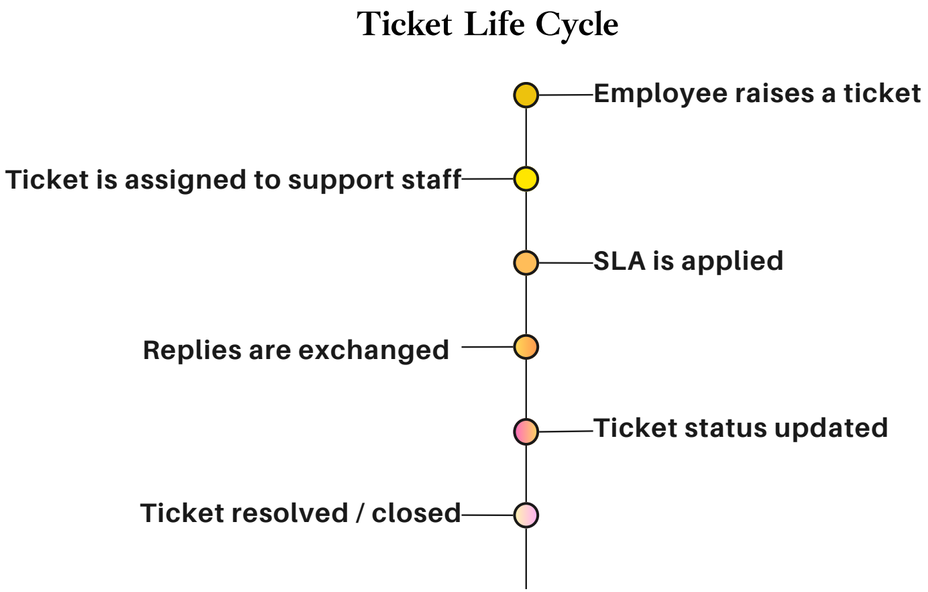
03. SLA

Method	Functionality
CreateSlaAsync()	Creates a new SLA policy
UpdateSlaAsync()	Updates SLA configuration
DeleteSlaAsync()	Deletes SLA if not linked to ticket types
GetSlaByIdAsync()	Fetches SLA by SLA ID
GetAllSlasAsync()	Retrieves all SLA records

04. Ticket Type

Method	Functionality
CreateTicketTypeAsync()	Creates a new ticket type
UpdateTicketTypeAsync()	Updates ticket type details
DeleteTicketTypeAsync()	Deletes ticket type if unused
GetTicketTypeByIdAsync()	Fetches ticket type by ID
GetAllTicketTypesAsync()	Retrieves all ticket types
GetByDepartmentIdAsync()	Retrieves ticket types by department
GetBySlaIdAsync()	Retrieves ticket types by SLA

05. Ticket



Method	Functionality
CreateTicketAsync()	Creates a new support ticket
UpdateTicketAsync()	Updates ticket status or assignment
DeleteTicketAsync()	Deletes ticket after validation
GetTicketByIdAsync()	Fetches ticket by ticket ID
GetAllTicketsAsync()	Retrieves all tickets
GetByEmpIdAsync()	Retrieves tickets created or assigned
GetByStatusAsync()	Retrieves tickets by status
GetByDepartmentIdAsync()	Retrieves tickets by department
GetByDepartmentAndStatusAsync()	Retrieves tickets by department and
GetByTicketTypeIdAsync()	Retrieves tickets by ticket type
GetOverdueTicketsAsync()	Retrieves overdue tickets based on

06. Ticket Reply

Method	Functionality
CreateTicketReplyAsync()	Adds a reply to ticket conversation
UpdateTicketReplyAsync()	Updates existing reply message
DeleteTicketReplyAsync()	Deletes a ticket reply
GetTicketReplyByIdAsync()	Fetches reply by reply ID
GetAllTicketRepliesAsync()	Retrieves all ticket replies
GetByTicketIdAsync()	Retrieves replies for a specific ticket
GetByEmployeeIdAsync()	Retrieves replies by employee

Future Scope & Challenges Faced

Future Scope:

- Email notifications
- File attachments in replies
- Ticket escalation
- Analytics dashboard
- Mobile application

Challenges Faced:

- Managing foreign key relationships
- Ensuring real-time UI updates
- Handling authentication tokens
- Implementing conversation flow correctly

Conclusion

- Successfully built a centralized ticket system
- Improved transparency and efficiency
- Reduced manual intervention
- Enhanced communication between employees and support teams

THANK YOU

We sincerely acknowledge S. N. Rao sir for his constant support and guidance throughout the successful completion of this project.

1. Abhishek Kumar M
2. Arnold Binu Varghese
3. Emi Mary Rose
4. Shwetanjali
5. Siranjeev M

Git Hub Link: https://github.com/ABHISHEK-KUMAR-M/Capestone-project