Sri Lanka Institute of Information Technology

GadgetBadget System

Programming Applications and Frameworks (IT 3030)

2020

Project ID: 09_PAF_Final_Project

Submitted By:

IT18126402	**Gamlath G.R.G.K.**
IT18021080	**Balasuriya D.A.M.**
IT18209358	**Silva A.H.D.M.**
IT18147674	**Perera W.T.H**
IT18203172	**Rajapaksha R.P.S.C.**

Contents

1. Members' Details and Workload Allocation

IT Number	Name	Web Service	Description of the Web Service
IT18126402	**Gamlath G.R.G.K.**	**Funding Buddies Services**	• Add a new Fund • Update Fund • Delete Fund • View all Fund
IT18021080	**Balasuriya D.A.M.**	**Order Services**	• Add a new Order • Update Order • Delete Order • View all Order
IT18209358	**Silva A.H.D.M.**	**Product(Research)Services**	• Add a new Product • Update Product • Delete Product • View all Product
IT18147674	**Perera W.T.H**	**Delivery Services**	• Add a new Delivery • Update Delivery • Delete Delivery • View all Delivery
IT18203172	**Rajapaksha R.P.S.C.**	**Payment Services**	• Add a new Payment • Update Payment • Delete Payment • View all Payment

2. Clickable link to GitHub Repository

https://github.com/GayanKavinda/09_PAF_Final-Project

3. SE Methodologies/Methods

V-Model

The V-Model is a type SDLC model where processes are executed in a sequential manner.it is an extension of waterfall mode.it is also called as verification and validation model.it is associated with a testing phase for each development stage and development phase is directly associated with testing phase. Development of the next phase starts only after the completion of previous phase.by using this model we gained lots of benefits to our project because we can correct the errors, bugs and test it before we enter the next stage. We can identify the defects in the early stages. It is better to identify the errors and bugs early as possible because when it comes to end of the project the team members have to put more effort and time to find and manage these errors. We applied this methodology to our project and gained the maximum out of it by reducing the disadvantages as much as we can.

The usage of V- Model

- More suitable for small to medium size projects
- Better to use when Software requirements are clear and well defined

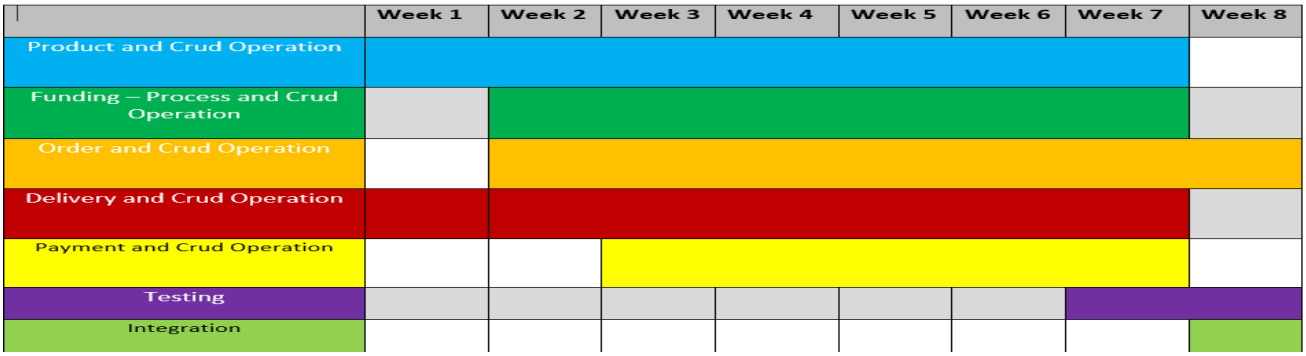
Advantages of V-Model

- Simple and easy to use
- Helps the project management to track progress of the work
- Highly disciplined model

Disadvantages of V-Model

- High risk and uncertainty
- Not suitable for complex projects
- Not suitable for projects where project requirements are not well defined

4. Gantt chart

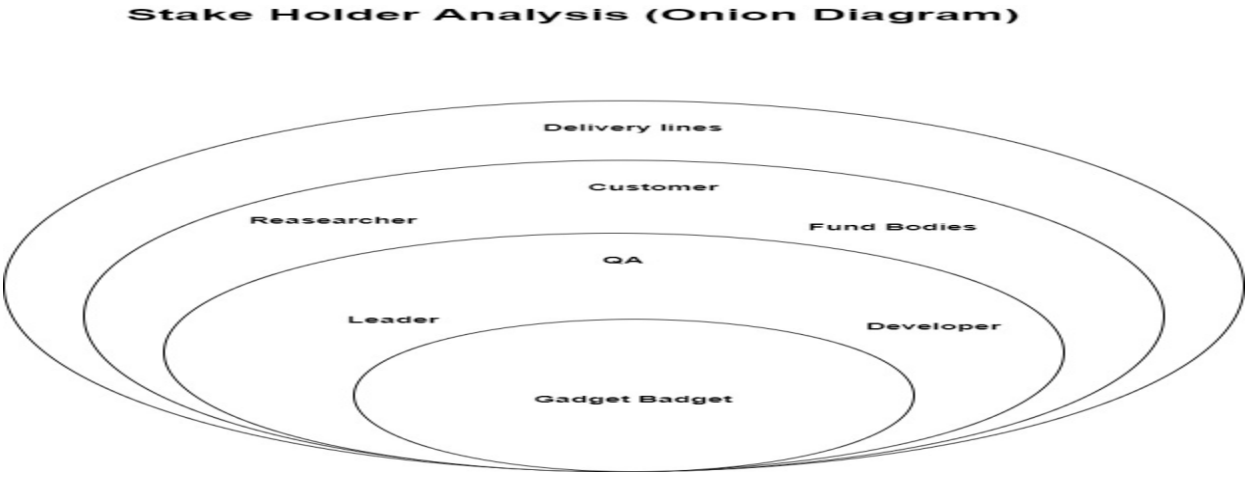


5. Requirements

5.1 Requirement analysis

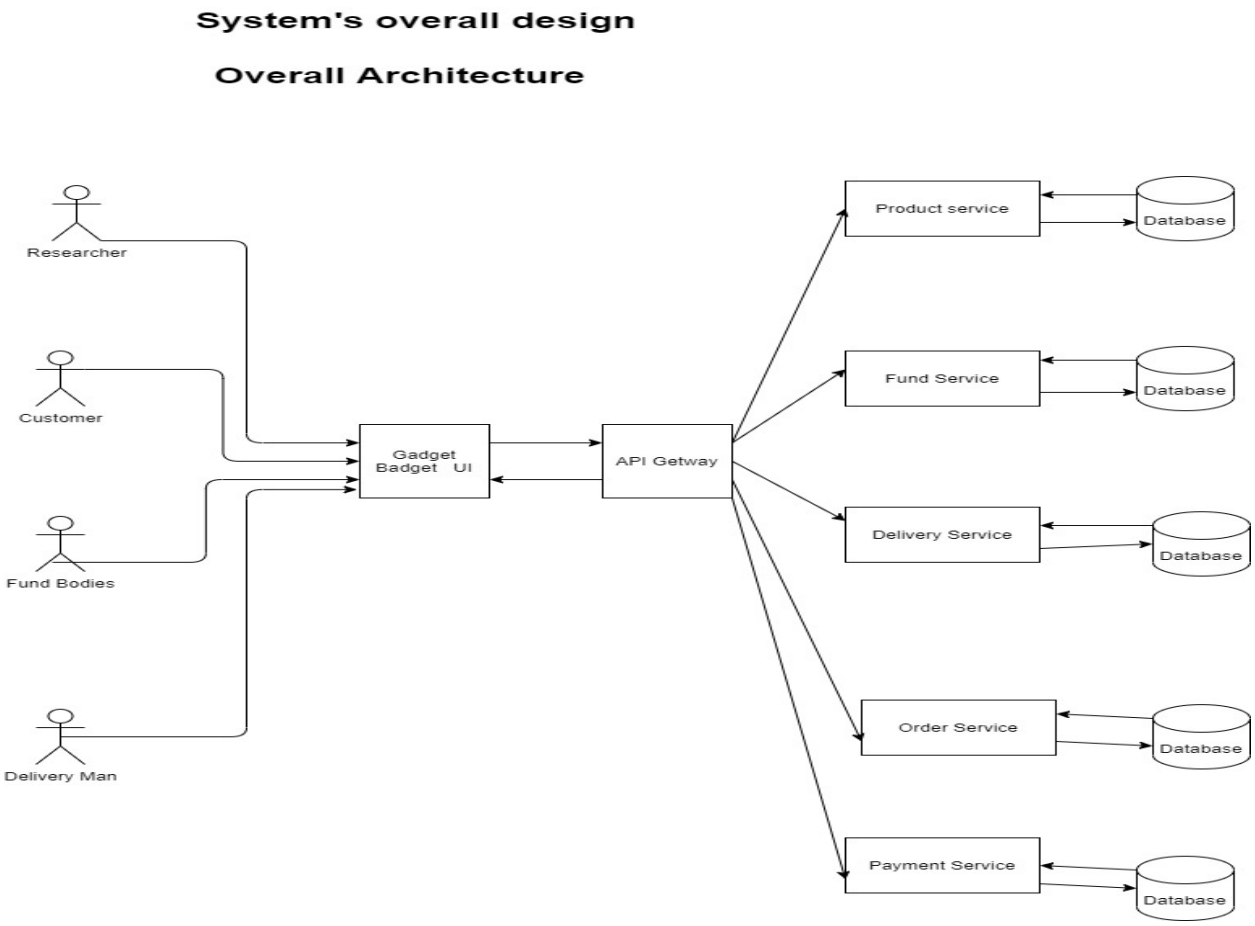
Micro-service	Functional Requirement	Non-Functional Requirement	Technical Requirement
Funding Buddies Services	• Add a new Fund • Update Fund • Delete Fund • View all Fund	Security, usability, performance	Funding person can insert, update, delete and view funds.
Order Services	• Add a new Order • Update Order • Delete Order • View all Order	Security, usability, performance	Customer can insert, update, delete and view Orders.
Product(Research)Services	• Add a new Product • Update Product • Delete Product • View all Product	Security, usability, performance	Researcher can insert, update, delete and view Products.
Delivery Services	• Add a new Delivery • Update Delivery • Delete Delivery • View all Delivery	Security, usability, performance	Delivery person can insert, update, delete and view Delivery details.
Payment Services	• Add a new Payment • Update Payment • Delete Payment • View all Payment	Security, usability, performance	Payment handler can insert, update, delete and view payment details.

5.2 Stake Holder Analysis (Onion Diagram)

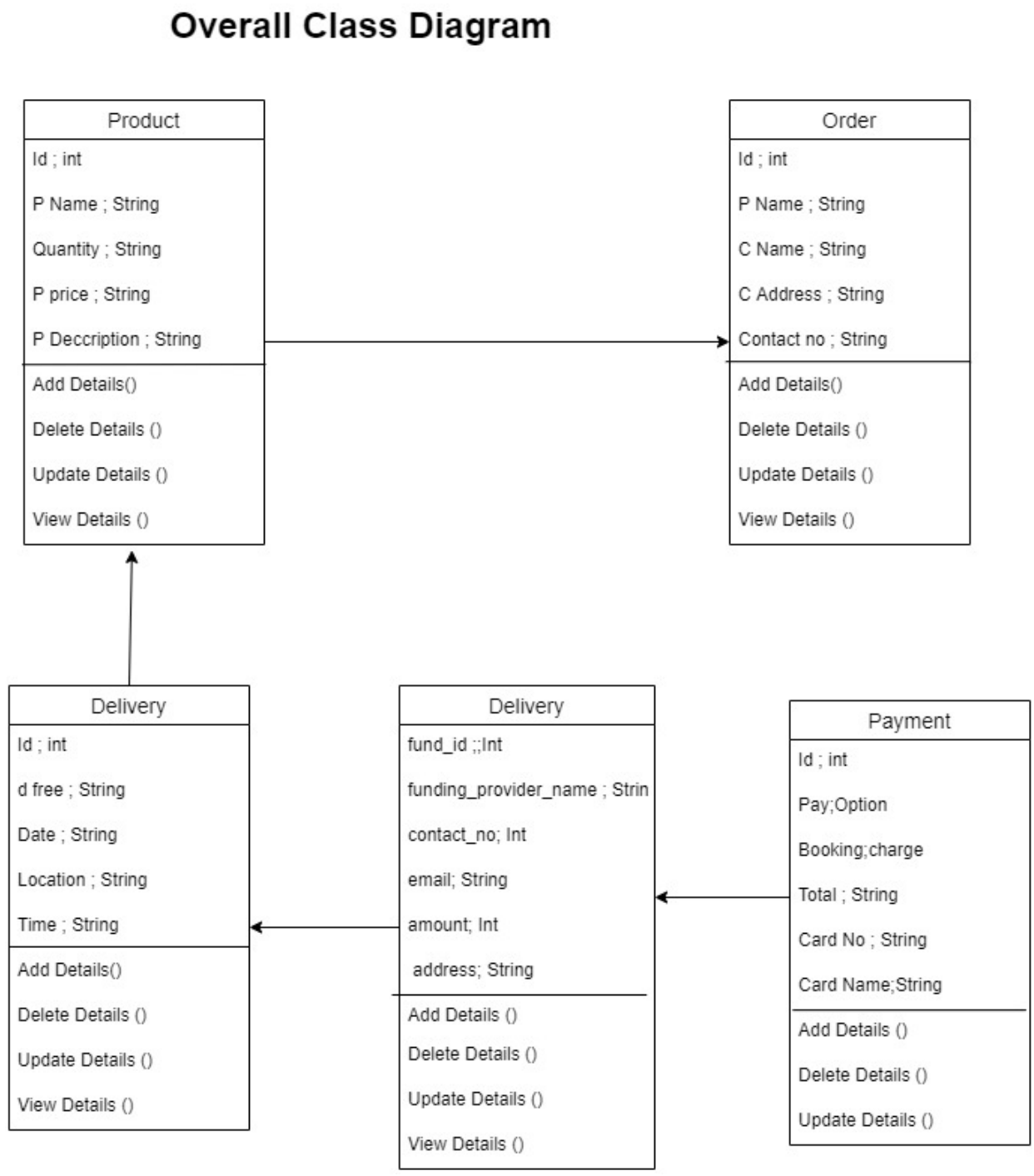


6. System's overall design

6.1 Overall Architecture



6.2 Class Diagram



7. Individual Section

IT18209358

Silva A.H.D.M.

Description

Gadget Budget system is a system which can perform several tasks such as make an The aim of the project is to develop a highly scalable online platform for the stakeholders of GB to interact with each other .The researcher can View their product can update, delete, and view product details.

Product Details

API for get all the Product details in the database (GET Request)

url :

Request : { pid = “auto generated integer value “ }

{ pid = “P001” }

{ pname = “hasini” }

{ rname = “Madhubhashini” }

{ quantity = “1” }

{ pprice = “20000” }

{pdescription=”new for list”}

Response: { pid = “auto generated integer value “ }

{ pid = “P002” }

{ pname = “Dilanka” }

{ rname = “Amarasinghe” }

{ quantity = “2” }

{ pprice = “30000” }

{pdescription=”new for list 2021”}

API for insert a new Product to the database (POST Request)

url :

Request :- { pid = “auto generated integer value “ }

{ pid = “P003” }

{ pname = “Dharmasena” }

{ rname = “Kumara” }

{ quantity = “3” }

{ pprice = “40000” }

{pdescription=”Today new list”}

Response :- { result = “Inserted successfully” }

{ p id = “auto generated integer value”}

{ error = “Error while inserting the product” }

API for update a patient which is existing in database (PUT Request)

url:

Request: { pid = “auto generated integer value “ }

{ pid = “P004” }

{ pname = “dasun” }

{ rname = “Pathirana” }

{ quantity = “4” }

{ pprice = “50000” }

{pdescription=”Today new list for the product”}

Response :- { result = “Updated successfully “}

{ error = “Error while updating the patient” }

API for delete a patient which is existing in database (DELETE Request)

url:

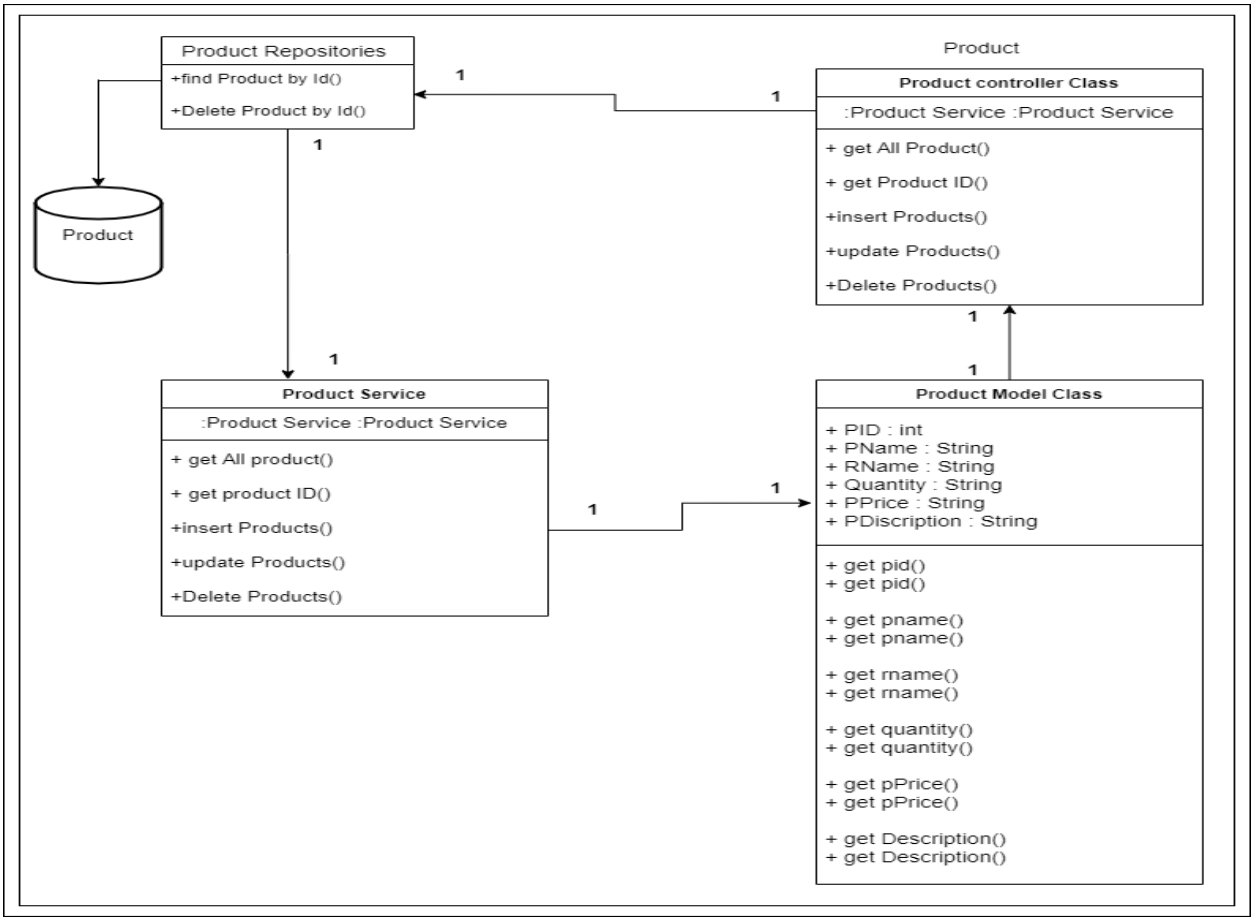
Request :- { p id = selected id from the service }

Response :- { result = “Deleted successfully” }

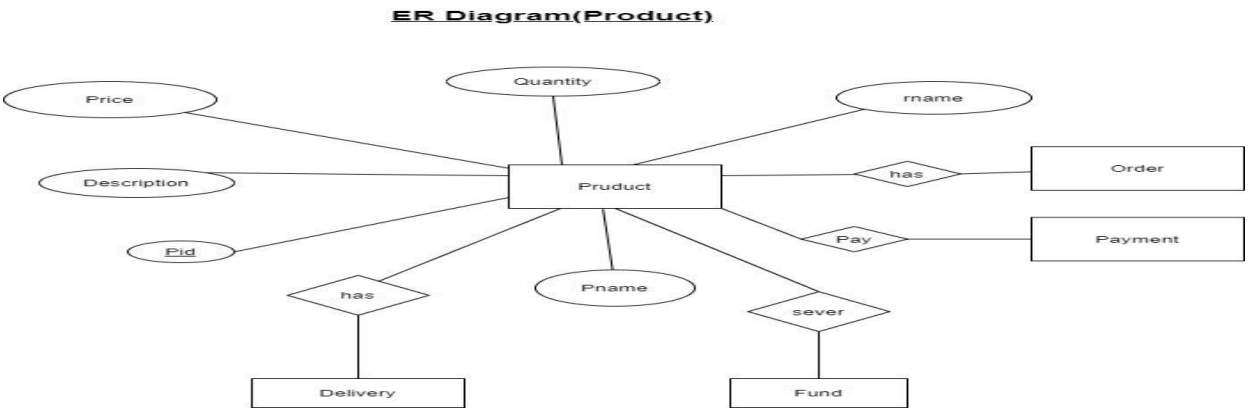
{ error = “Error while deleting the product” }.

Service Design

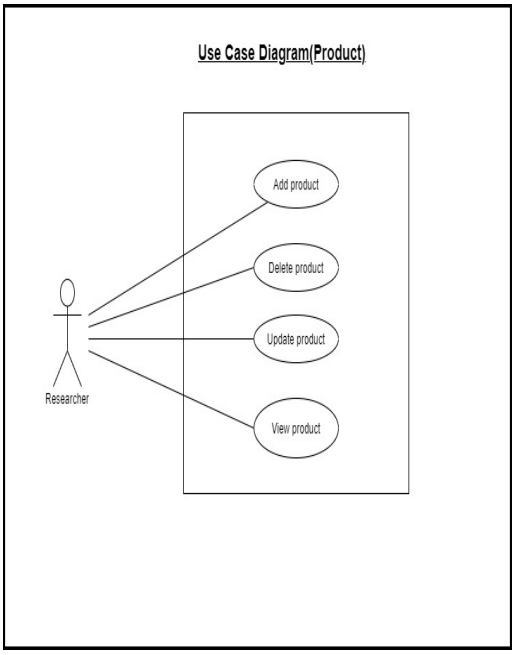
Class Diagram-



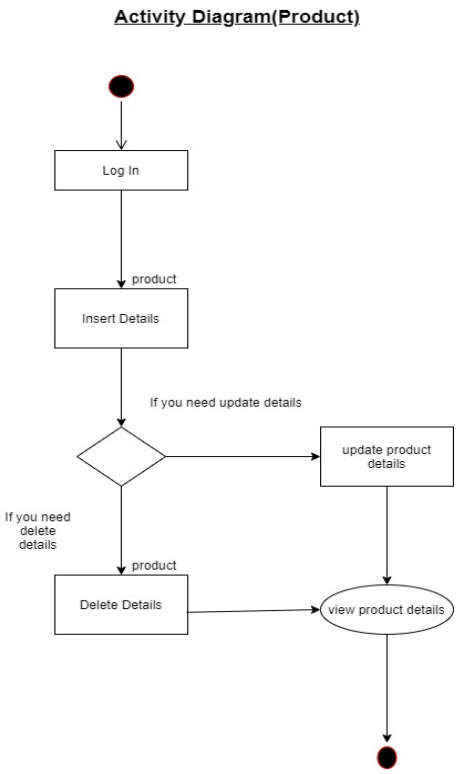
ER Diagram



Use case Diagram



Activity Diagram



IT18147674

Perera W.T.H

Description

Gadget Budget system is a system which can perform several tasks such as make an The aim of the project is to develop a highly scalable online platform for the stakeholders of GB to interact with each other .The delivery man can View their product can update, delete, and view delivery details.

Delivery Details

API for get all the Product details in the database (GET Request)

url:

Request :- { did = “auto generated integer value “ }

{ did = “P001” }

{ pfee = “500” }

{ ddate = “21/03/2021” }

{ location = “56/7,kurunegala road,Bingiriya” }

{ time= “2.00 P.M.” }

Response :- { result = “Inserted successfully” }

{ did = “auto generated integer value”}

{ error = “Error while inserting the delivery details” }

Request :- { did = “auto generated integer value “ }

{ did = “P002” }

{ pfee = “570” }

{ ddate = “30/04/2021” }

{ location = “56/7,kurunegala road,Medagama” }

{ time= “7.00 P.M.” }

Response :- { result = “Inserted successfully” }

{ did = “auto generated integer value”}

{ error = “Error while inserting the delivery details” }

API for insert a new Product to the database (POST Request)

url:

Request :- { did = “auto generated integer value “ }

{ did = “d004” }

{ pfee = “1570” }

{ ddate = “30/009/2021” }

{ location = “78,kurunegala road,Medagama” }

{ time= “1.00 P.M.” }

Response :- { result = “Inserted successfully” }

{ did = “auto generated integer value”}

{ error = “Error while inserting the Delivery deyails” }

API for update a patient which is existing in database (PUT Request)

url :

Request :- { did = “auto generated integer value “ }

{ did = “d005” }

{ pfee = “2570” }

{ ddate = “30/10/2021” }

{ location = “No2,kurunegala road,Kurunegala” }

{ time= “9.00 A.M.” }

Response :- { result = “Updated successfully “}

{ error = “Error while updating the product” }

API for delete a patient which is existing in database (DELETE Request)

url :

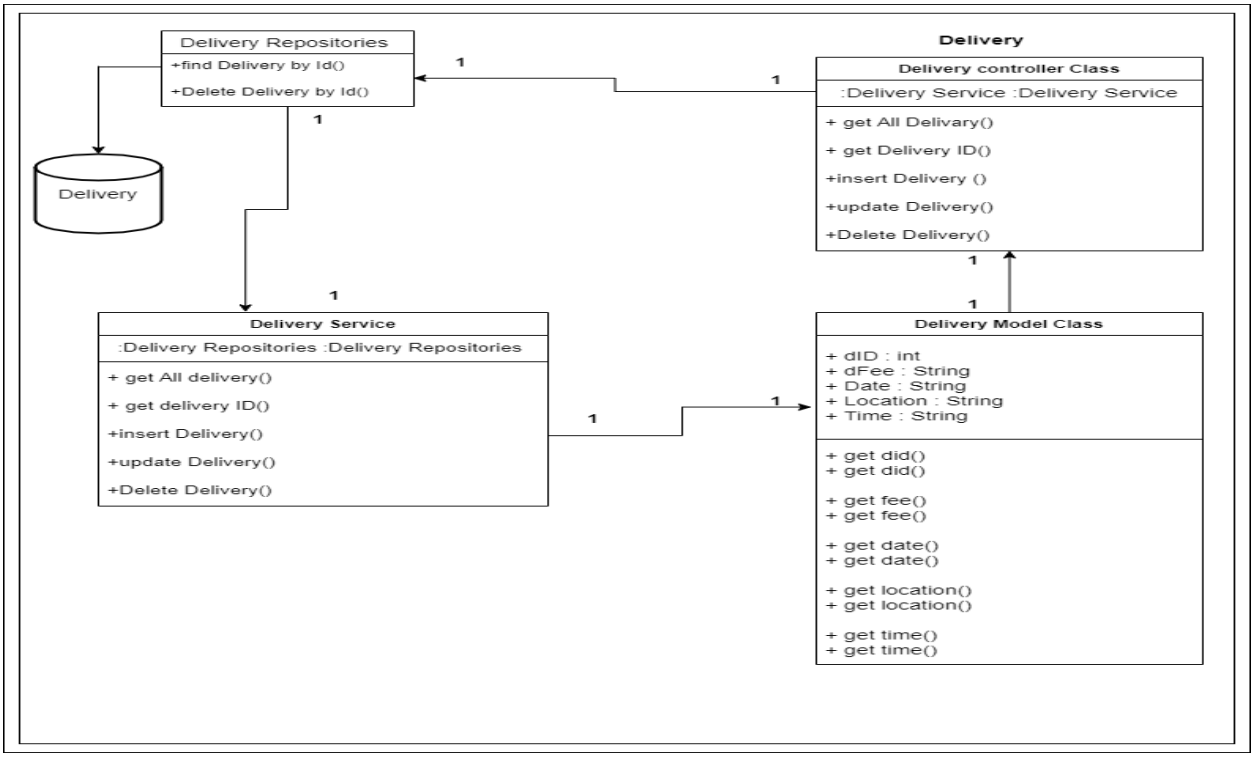
Request :- { did = selected id from the service }

Response :- { result = “Deleted successfully” }

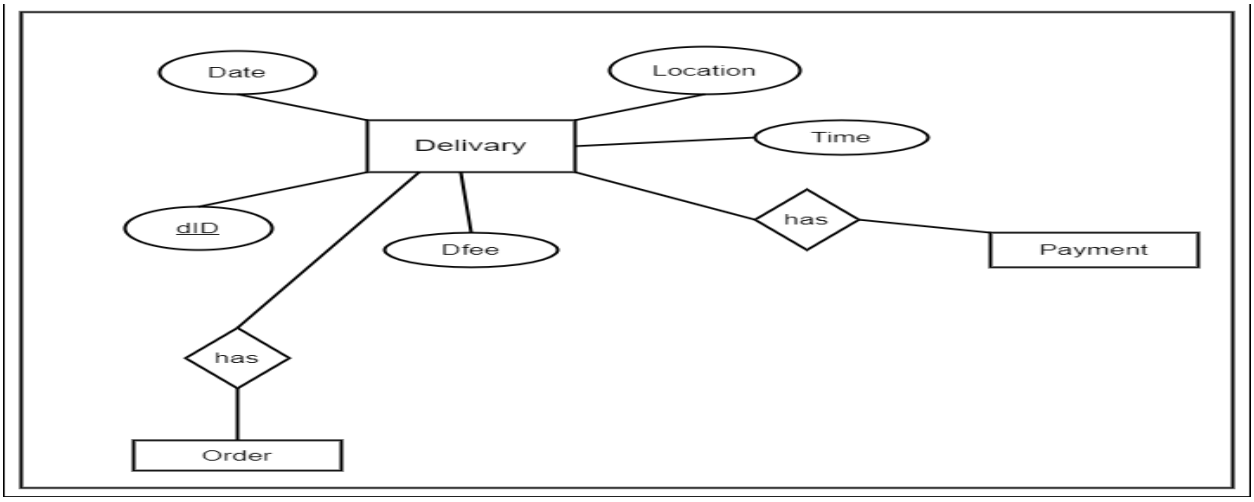
{ error = “Error while deleting the Delivery Details” }.

Service Design

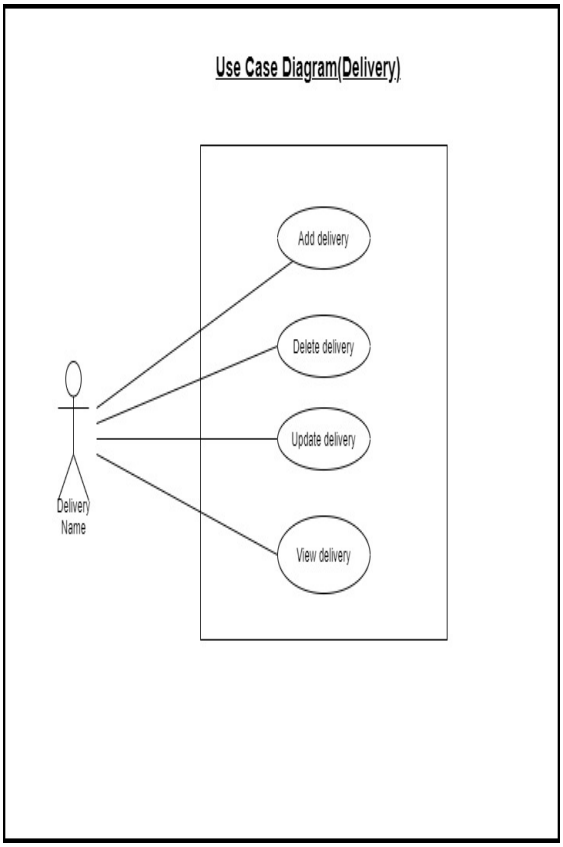
Class Diagram



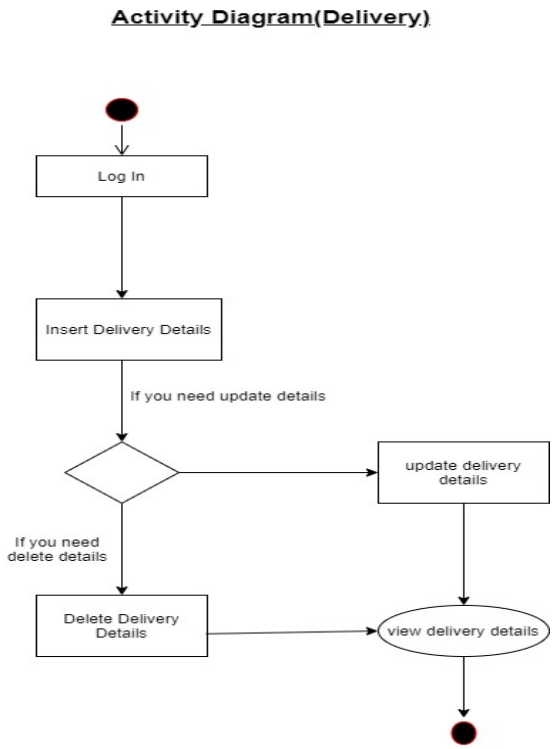
ER diagram



Use Case Diagram



Activity Diagram



IT18126402

Gamlath G.R.G.K.

Description

Gadget Budget system is a system which can perform several tasks such as make an The aim of the project is to develop a highly scalable online platform for the stakeholders of GB to interact with each other .The Fund Persons can View their product can update, delete, and view Fund details.

Fund Details

API for get all the Product details in the database (GET Request)

URL:

Request : { fund_id = “auto generated integer value “ }

{ fund_id = “F001” }

{ f_name = “Gayan” }

{ contact_no= “0789875632” }

{ email = “gayan@gmail.com” }

{ amount = “20000” }

{address=”N02:Gampaha.”}

Response: { fund_id = “auto generated integer value “ }

{ fund_id = “F002” }

{ f_name = “Akalanka” }

{ contact_no= “0789456832” }

{ email = “gayan12@gmail.com” }

{ amount = “10000” }

{address=”N02:Gampaha,Kotte.”}

API for insert a new Product to the database (POST Request)

url:

Request :- { fund_id = “auto generated integer value “ }

{ fund_id = “F003” }

{ f_name = “Gayan” }

{ contact_no= “0789875632” }

{ email = “gayan@gmail.com” }

{ amount = “20000” }

{address=”N02:Gampaha.”}

Response :- { result = “Inserted successfully” }

{ p id = “auto generated integer value”}

{ error = “Error while inserting the Funding” }

API for update a patient which is existing in database (PUT Request)

url:

Request :- { fund_id = “auto generated integer value “ }

{ fund_id = “F001” }

{ f_name = “Gayan” }

{ contact_no= “0789875632” }

{ email = “gayan@gmail.com” }

{ amount = “20000” }

{address=”N02:Gampaha.”}

Response :- { result = “Updated successfully “}

{ error = “Error while updating the funding” }

API for delete a patient which is existing in database (DELETE Request)

url:

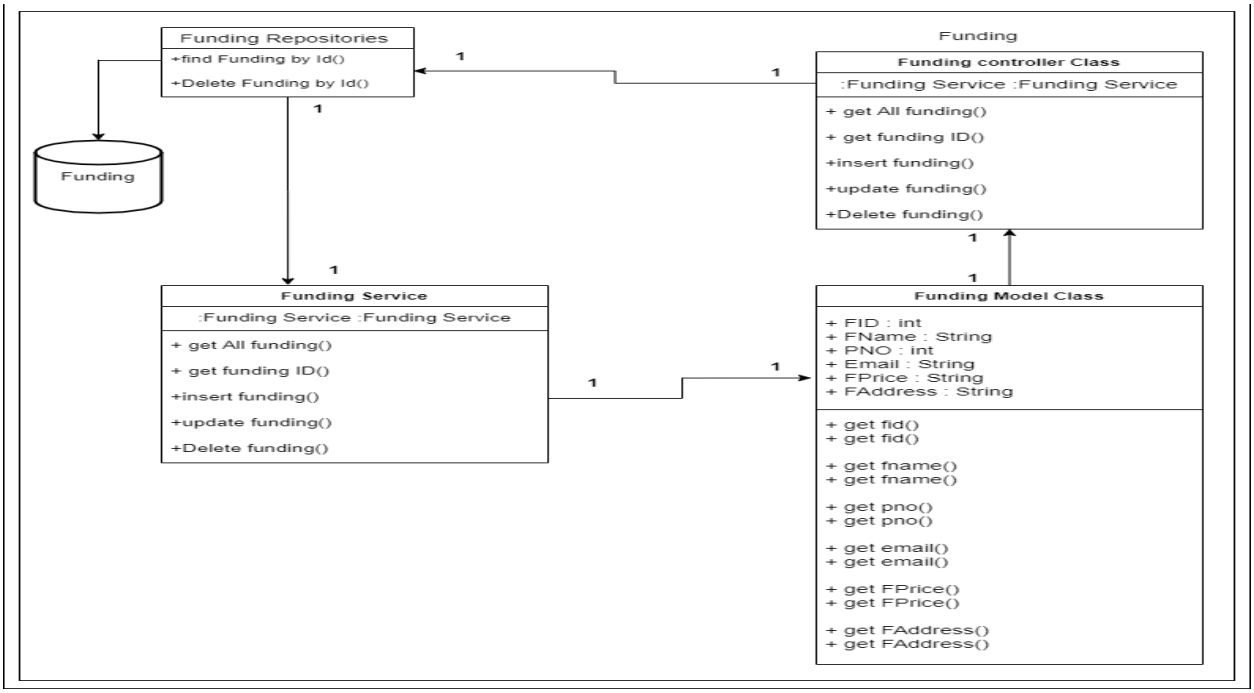
Request :- { p id = selected id from the service }

Response :- { result = “Deleted successfully” }

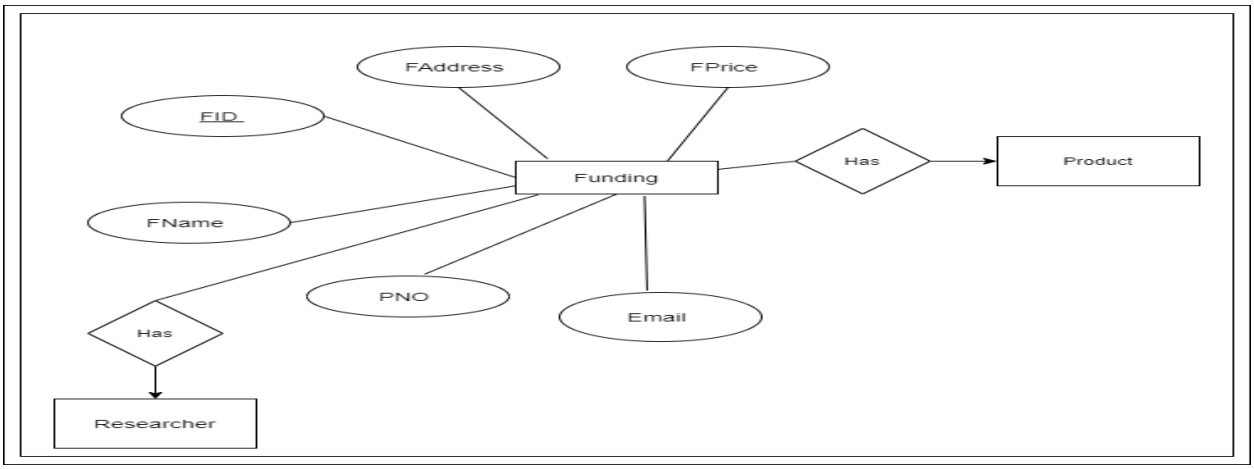
{ error = “Error while deleting the Funding Details” }.

Service Design

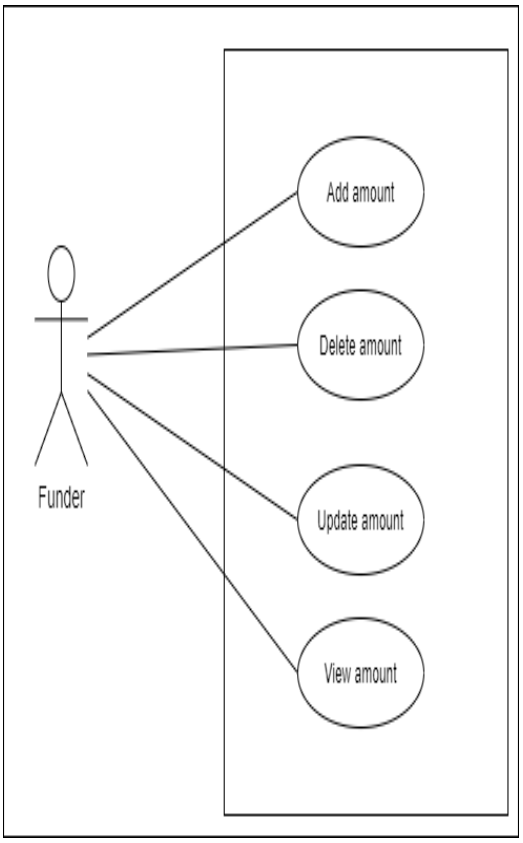
Class Diagram



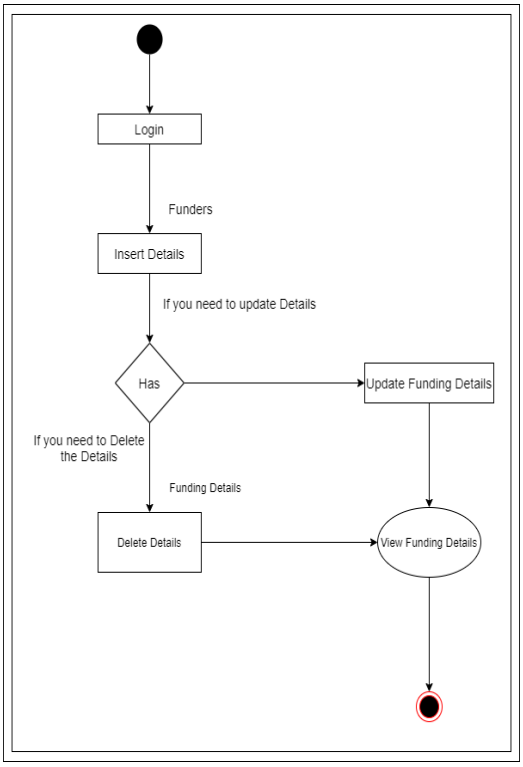
ER Diagram



Use Case Diagram



Activity Diagram



IT18021080

Balasuriya D.A.M.

Description

Gadget Budget system is a system which can perform several tasks such as make an The aim of the project is to develop a highly scalable online platform for the stakeholders of GB to interact with each other .The Customers can View their product can update, delete, and view Order details.

Order Details

API for get all the Product details in the database (GET Request)

URL:

Request : { order_id = “auto generated integer value “ }

{ order_id = “001” }

{ p_name = “Minosh” }

{ c_name= “Balasuriya” }

{ c_address = “56/4Kurunegala road,Chilaw” }

{ contactNo = “0763589628” }

Response: { order_id = “auto generated integer value “ }

{ order_id = “002” }

{ p_name = “Nimantha” }

{ c_name= “Fernando” }

{ c_address = “No2:Kurunegala road,Kurunegala” }

{ contactNo = “0763534576” }

API for insert a new Product to the database (POST Request)

url:

Request :- { order_id = “auto generated integer value “ }

{ order_id = “004” }

{ p_name = “Kasun” }

{ c_name= “Perera” }

{ c_address = “No34:Negambo” }

{ contactNo = “0987867324” }

Response :- { result = “Inserted successfully” }

{ p id = “auto generated integer value”}

{ error = “Error while inserting the Order” }

API for update a patient which is existing in database (PUT Request)

url:

Request :- { order_id = “auto generated integer value “ }

{ order_id = “005” }

{ p_name = “Minol” }

{ c_name= “Prageesha” }

{ c_address = “No”02 Thewatha.” }

{ contactNo = “07555508231” }

Response :- { result = “Updated successfully “}

{ error = “Error while updating the Ordering” }

API for delete a patient which is existing in database (DELETE Request)

url:

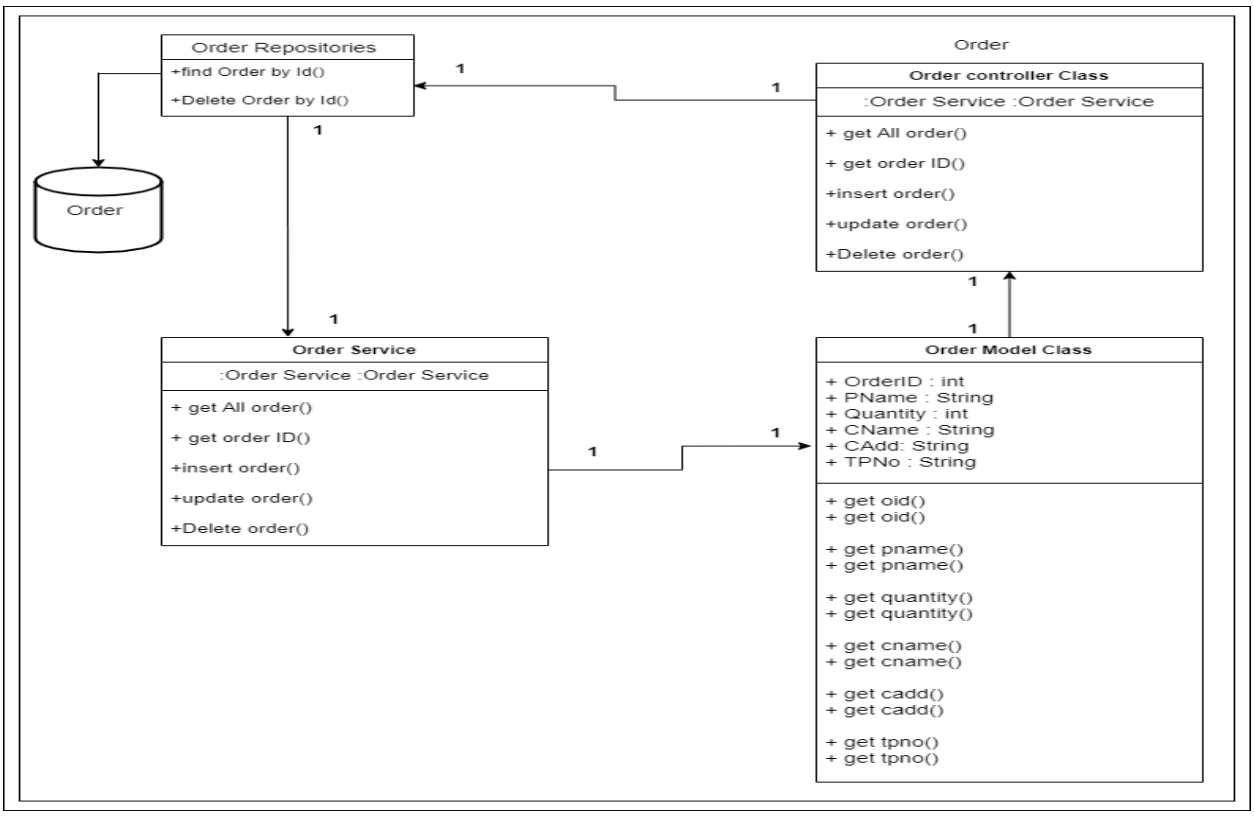
Request :- { p id = selected id from the service }

Response :- { result = “Deleted successfully” }

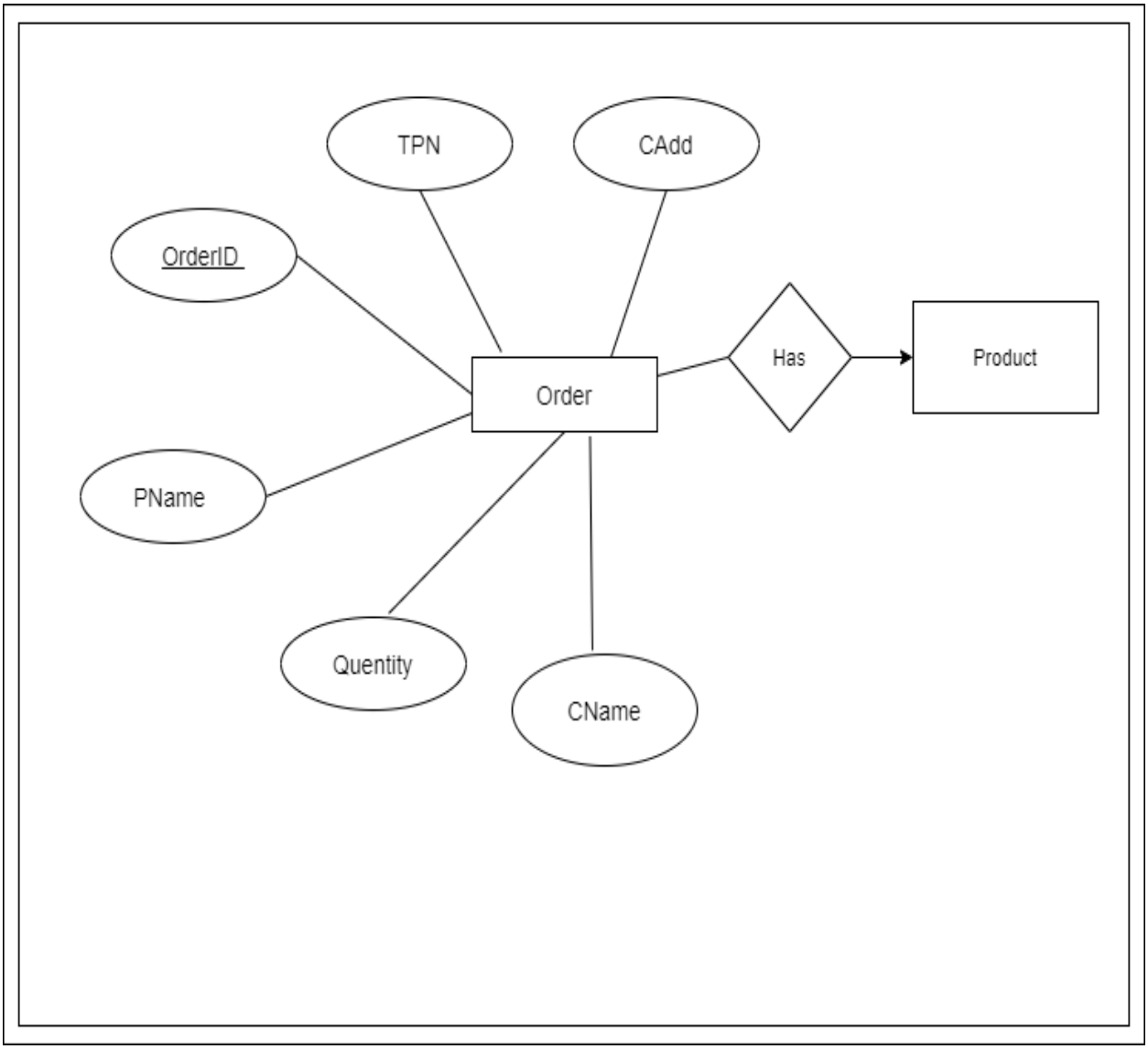
{ error = “Error while deleting the Order Details” }.

Service Design

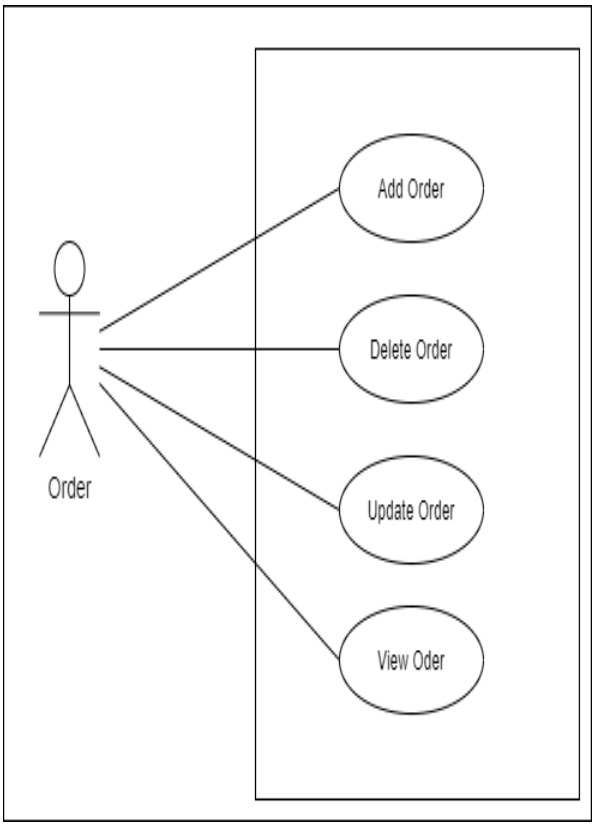
Class Diagram



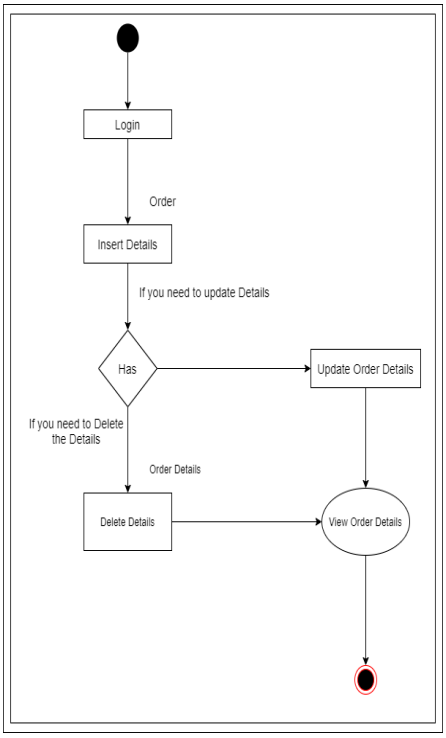
ER Diagram



Use Case Diagram



Activity Diagram



IT18203172

Rajapaksha R.P.S.C.

Description

Gadget Budget system is a system which can perform several tasks such as make an The aim of the project is to develop a highly scalable online platform for the stakeholders of GB to interact with each other .The Payment Handler can View their product can update, delete, and view Payment details.

Payment Details

API for get all the Product details in the database (GET Request)

URL:

Request : { payment_id = “auto generated integer value “ }

{ payment_id = “001” }

{ name_on_card = “Commercial” }

{ cvc= “777” }

{ exp_day= “03/09/2023” }

Response: { payment_id = “auto generated integer value “ }

{ payment_id = “002” }

{ name_on_card = “Sampath” }

{ cvc= “776” }

{ exp_day= “03/05/2024” }

API for insert a new Product to the database (POST Request)

url:

Request :- { payment_id = “auto generated integer value “ }

{ payment_id = “003” }

{ name_on_card = “Seylanl” }

{ cvc= “778” }

{ exp_day= “03/09/2024” }

Response :- { result = “Inserted successfully” }

{ p id = “auto generated integer value”}

{ error = “Error while inserting the payment” }

API for update a patient which is existing in database (PUT Request)

url:

Request :- { payment_id = “auto generated integer value “ }

{ payment_id = “004” }

{ name_on_card = “nationsTrust” }

{ cvc= “444” }

{ exp_day= “03/09/2023” }

Response :- { result = “Updated successfully “}

{ error = “Error while updating the Payment” }

API for delete a patient which is existing in database (DELETE Request)

url:

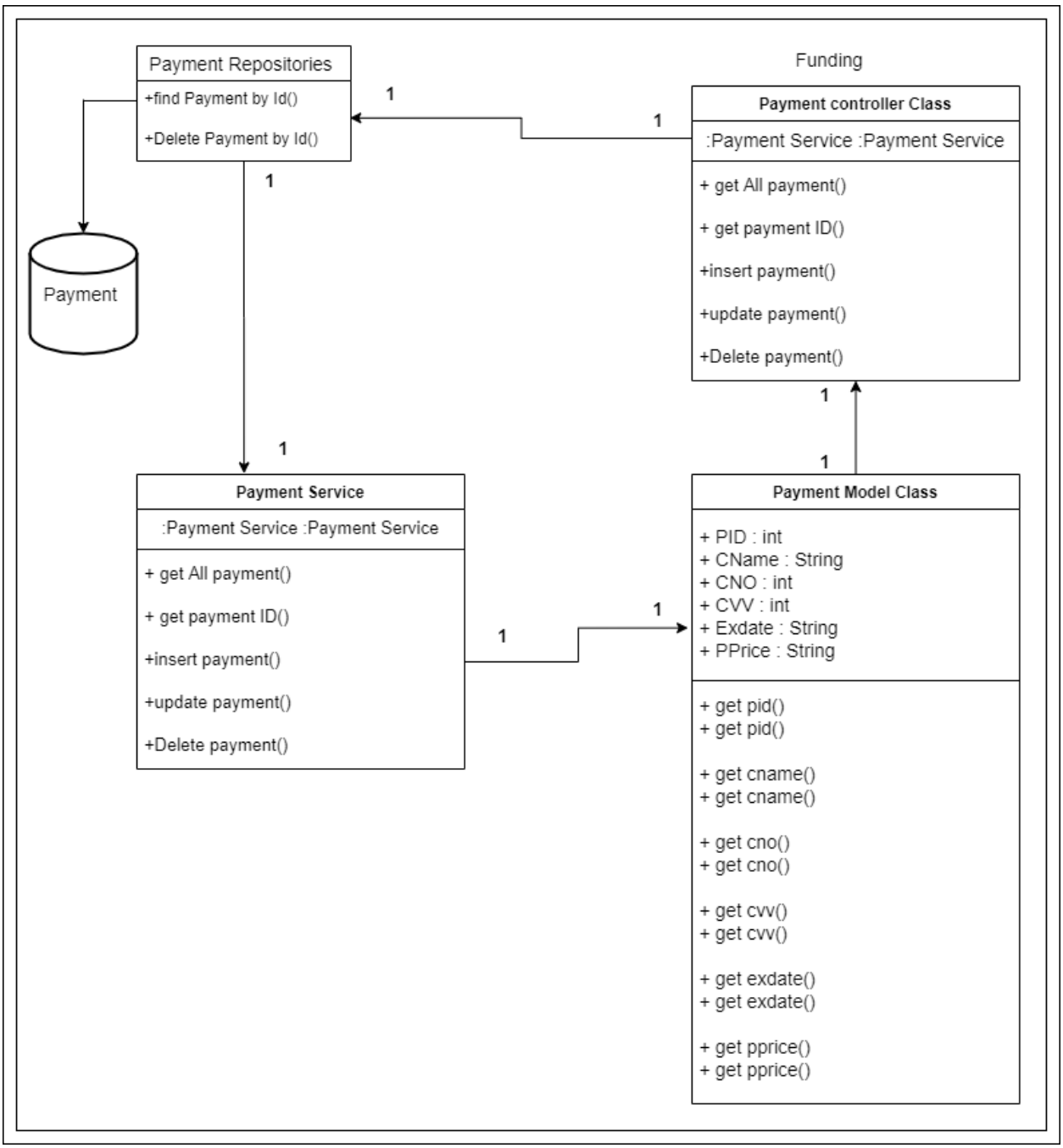
Request :- { p id = selected id from the service }

Response :- { result = “Deleted successfully” }

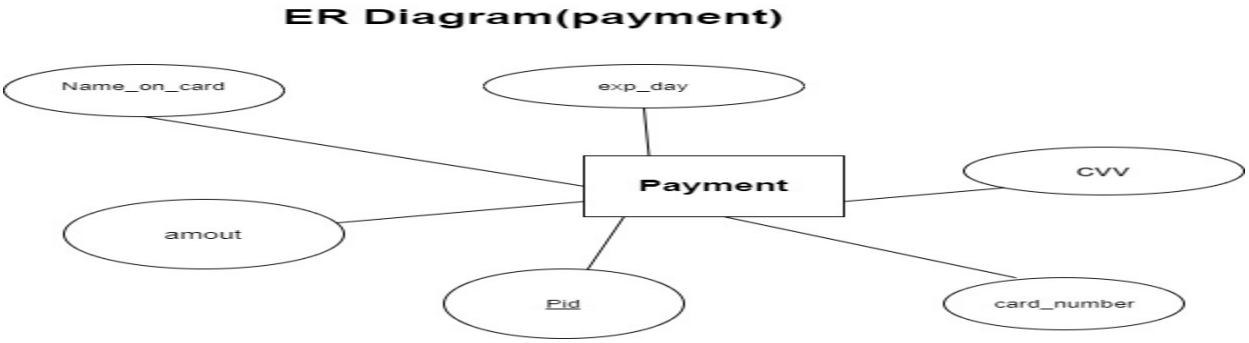
{ error = “Error while deleting the Paymentt Details” }.

Service Design

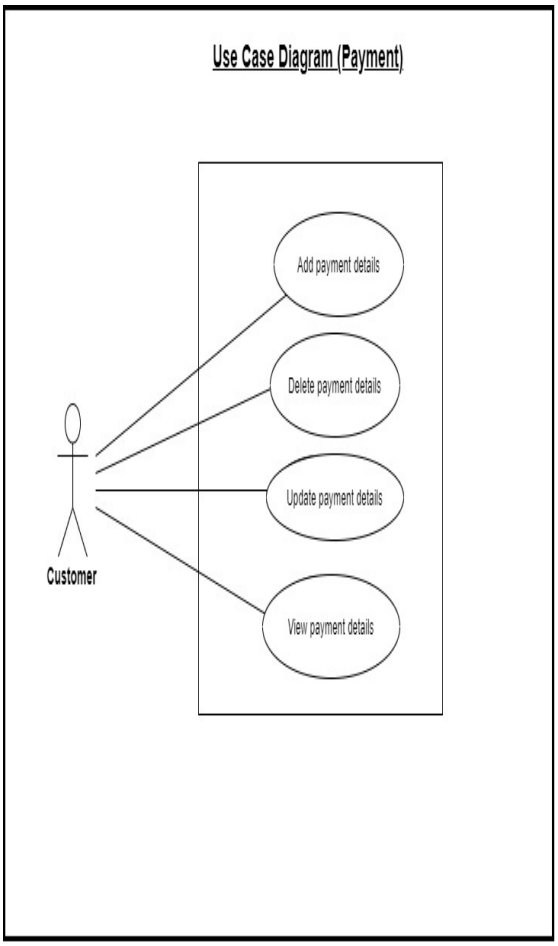
Class Diagram



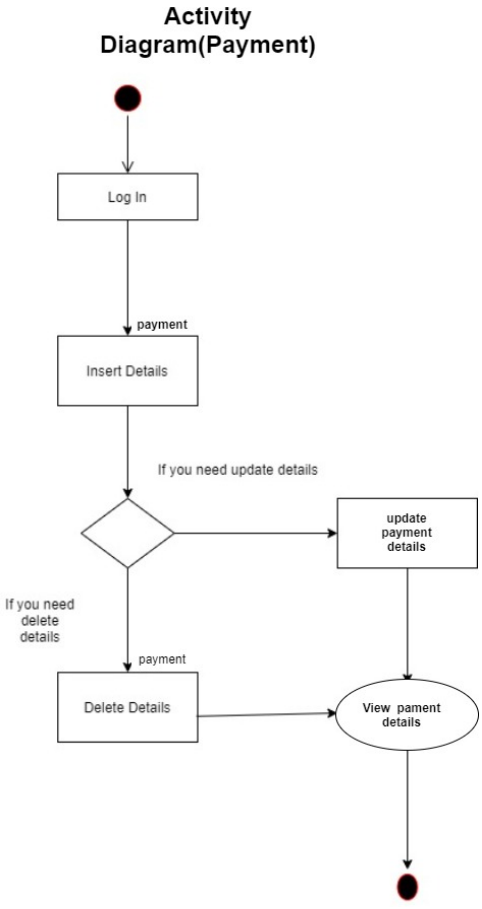
ER Diagram



Use Case Diagram



Activity Diagram



Appendix

Commit Logs of Team Members

IT18126402 **.**	https://github.com/GayanKavinda/09_PAF_Final-Project/tree/Funding-Process
IT18021080	https://github.com/GayanKavinda/09_PAF_Final-Project/tree/Order
IT18209358	https://github.com/GayanKavinda/09_PAF_Final-Project/tree/Product
IT18147674	https://github.com/GayanKavinda/09_PAF_Final-Project/tree/Delivery
IT18203172	https://github.com/GayanKavinda/09_PAF_Final-Project/tree/Payment