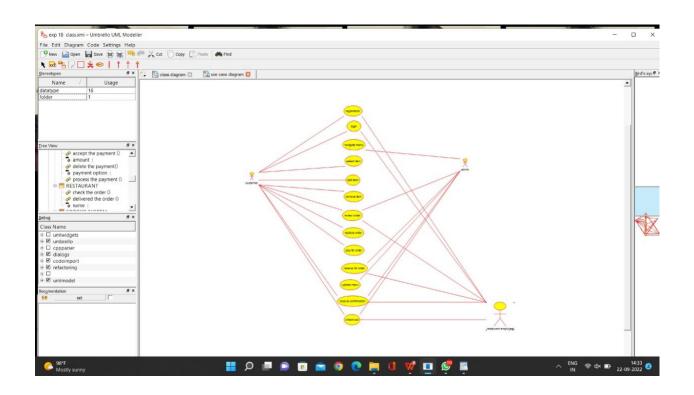
Name: B. Aravinda Valli

Reg No:192011442

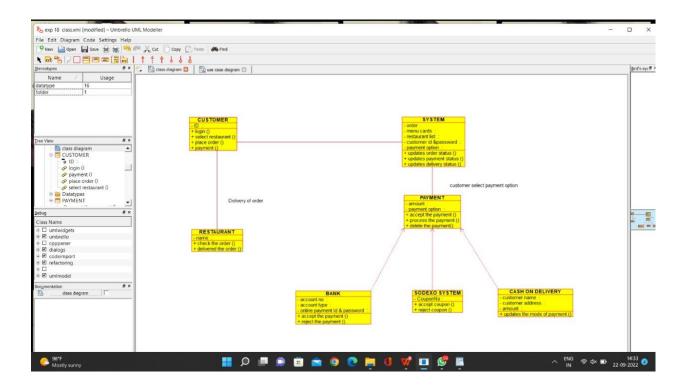
Experiment-18

Food order system

USE CASE DIAGRAM:



CLASS DIAGRAM:



#

coding=Sy

stem

from

PAYMENT

import *

шш		
:version:		
:author:		
ппп		
шш		
ATTRIBUT		
ES		
account_		
no_		
(private)		

class

ect):

BANK(obj

ype_
(private)
#
coding=Sy
stem
from
PAYMENT
import *
class
CASH_ON
_DELIVER
Y(object):
шш
:version:
:author:
"""

account_t

ATTRIBUT ES
customer _name_ (private)
customer _address _
(private)

amount_

(private)

0.0010

```
.....
```

def updates_t he_mode _of_paym ent_(self) 111111 @return @author pass coding=Sy stem from SYSTEM import *

from
RESTAUR
ANT
import *
class
CUSTOME
R(object):
ппп
:version:
:author:
шш
шш
ATTRIBUT
ES

```
ID_
(private)
 .....
def
login_(sel
f):
  111111
@return
@author
  .....
  pass
def
select_res
taurant_(
self):
```

111111

@return
:

@author

pass

def
place_ord
er_(self):

"""

@return

:

@author

111111

```
pass
def
payment_
(self):
  111111
@return
@author
  .....
  pass
#
coding=Sy
stem
from
PERMISSI
```

ON_CLAS

S import

class
CUSTOME
R_CLASS
(PERMISSI
ON_CLAS
S):
ппп
:version:
:author:
нин
ппп
ATTRIBUT
ES

ID

(private)

```
mobile_a
nd_addre
ss
(private)

"""

def
add_cust
```

omer_(sel

f):

111111

name

@return
:
@author
нин
pass
def
edit_cust
omer_(sel
f):
@return
:
@author
ини
pass

```
def
delete_cu
stomer_(s
elf):
 .....
@return
@author
 111111
 pass
def
search_cu
stomer_(s
elf):
 111111
```

@author			
pass			
#			
coding=Sy			
stem			
class			
DRIVING_			
LICENSE_			
CLASS(obj			
ect):			
ппп			
:version:			
:author:			

@return

11111	
""" ATTRIBUT ES	
type_ (private)	
name_ (private)	

ID (private)

.....

```
add_drivi
ng_licens
e_(self):
 .....
@return
@author
 111111
 pass
def
edit_drivi
ng_licens
e_(self):
```

111111

def

```
@return
@author
 .....
 pass
def
delete_dr
iving_lice
nse_(self)
 111111
@return
@author
 111111
 pass
```

```
def
search\_dr
iving_lice
nse_(self)
  .....
@return
@author
  111111
  pass
#
coding=Sy
stem
class
DRIVING_
LICENSE_
CLASS(obj
ect):
```

:version:
:author:
шш
шш
ATTRIBUT
ES
type_
(private)
name_

(private)

ID (private) def add_drivi ng_licens e_(self): @return @author

pass

```
def
edit_drivi
ng_licens
e_(self):
  .....
@return
@author
  111111
  pass
def
delete_dr
iving_lice
nse_(self)
  111111
```

```
@author
 .....
 pass
def
search_dr
iving_lice
nse_(self)
 111111
@return
@author
 111111
 pass
```

@return

#
coding=Sy
stem
class
PERMISSI
ON_CLAS
S(object):
шш
:version:
:author:
.uutiioi.
ппп
ATTRIBUT
ES

```
ID_
(private)
role_
(private)
title_
(private)
 .....
def
add_per
mission_(
self):
```

111111

```
@return
@author
 .....
 pass
def
edit_per
mission_(
self):
 .....
@return
@author
  .....
 pass
```

```
def
delete_pe
rmission_
(self):
  .....
@return
@author
  111111
  pass
def
search_p
ermission \\
_(self):
  111111
```

@author pass coding=Sy stem from PERMISSI ON_CLAS S import class REGISTRA TION_CLA SS (PERMISSI ON_CLAS S):

111111

@return

:author:			
ATTRIBUT			
ES			
ID_			
(private)			
customer			
(private)			

:version:

```
number_
(private)
 name
(private)
type_
(private)
 .....
def
add_regis
tration_(s
elf):
  111111
```

@return
:
@authar
@author
pass
def
edit_regis
tration_(s
elf):
Out to our
@return
:
@author
11111
pass

```
def
delete_re
gistration
_(self):
  .....
@return
@author
  .....
  pass
def
search_re
gistration
_(self):
  111111
```

@author			
нин			
pass			
#			
coding=Sy			
stem			
class			
ROLE_CL			
ASS(objec			
t):			
:version:			
version.			
:author:			

@return

""" ATTRIBUT ES

(private)

ID

title_ (private)

descriptio

(private)

n_

.....

```
def
add_role_
(self):
  .....
@return
@author
  111111
  pass
def
edit_role
_(self):
```

111111

@return
:
@author
11111
pass
def
delete_ro
le_(self):
шш
@return
:
@author
ппп
pass

```
def
search_ro
le_(self):
  111111
@return
@author
  111111
  pass
def
assign_rol
e_(self):
  .....
```

@return

@author
pass
#
coding=Sy
stem
class
USER(obj
ect):
11111
:version:
:author:
шш
пии
ATTRIBUT

ES

ID_ (private) def name_(se lf): @return @author pass

```
def
email_(se
lf):
  111111
@return
@author
  111111
  pass
def
password
_(self):
  .....
```

@return

:

```
@author
 .....
 pass
def
Cv_uploa
d_(self):
 .....
@return
@author
 111111
 pass
def
mobile_n
umber_(s
```

elf):

@return : @author 111111 pass # coding=Sy stem from PERMISSI ON_CLAS S import class VEHICLE_ CLASS(obj

ect):

:version:	
:author:	
шш	
ппп	
ATTRIBUT	
ES	
Id	
(private)	
type_	
(private)	

```
number_
(private)
customer
_id_
(private)
 .....
def
add_vehi
cle_(self):
  .....
```

@return

```
@author
 .....
 pass
def
edit_vehi
cle_(self):
 .....
@return
@author
 .....
 pass
def
delete_ve
hicle_(self
```

):

@return

:

@author

111111

pass

def

search_ve

hicle_(self

):

111111

@return

•

@author

```
111111
```

pass

#

coding=Sy

stem

from

PERMISSI

ON_CLAS

S import

*

class

VEHICLE_

TYPE_CLA

SS

(PERMISSI

ON_CLAS

S):

.....

#

coding=Sy

stem

class
PAYMENT
(object):
:version:
:author:
.autiloi .
ппп
ATTRIBUT
ES
£3
amount_
(private)

```
payment_
{\sf option}\_
(private)
 .....
def
accept\_th
e_payme
nt_(self):
  111111
@return
@author
  111111
  pass
```

```
def
process_t
he_paym
ent_(self)
 .....
@return
@author
 111111
 pass
def
delete_th
e_payme
nt(self):
```

@author	
шш	
pass	
#	
coding=Sy	
stem	
class	
RESTAUR	
ANT(obje	
ct):	
нин	
:version:	
:author:	
нин	

@return

```
111111
ATTRIBUT
ES
name_
(private)
 .....
def
check_th
e_order_(
self):
  .....
@return
```

@author

```
pass
```

def

delivered

_the_ord

er_(self):

.....

@return

:

@author

111111

pass

#

coding=Sy

stem

from

PAYMENT

import *

11111		
:version:		
:author:		
шш		
нин		
ATTRIBUT		
ES		
CouponN		
0		
(private)		

class

SODEXO_

SYSTEM(o

bject):

.....

def

accept_co

upon_(sel

f):

.....

@return

:

@author

.....

pass

def

reject_co

upon_(sel

f):

.....

@return :
@author
pass
#
coding=Sy
stem
from
PAYMENT
import *
class
SYSTEM(o
bject):

:version:

:author:
ппп
ATTRIBUT ES
E5
order_
(private)
menu_car
ds_
(private)

restauran

```
t_list_
(private)
customer
_id_pass
word_
(private)
payment_
option_
(private)
 111111
def
updates_
order_sta
tus_(self):
```

@return

:

@author

111111

pass

def

updates_

payment_

status_(s

elf):

111111

@return

.

@author

```
111111
```

pass

def

updates_

delivery_s

tatus_(sel

f):

111111

@return

:

@author

111111

pass

```
online_pa
yment_Id
_passwor
\mathsf{d}_{-}
(private)
 .....
 def
accept_th
e_payme
nt_(self):
  .....
@return
@author
  111111
  pass
```

def
reject_th
e_payme
nt_(self):
"""

@return

•

@author

111111

pass