Software methodology to be used: Incremental Development.

**Context model:**

Click Register

Fill registration form

Error message

Generate unique user id

Update user database

Display staff menu

Display food menu

Confirm order decision

Pay

Order & receipt generated

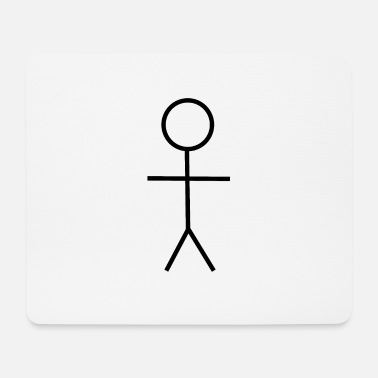
**Process Model:** 1st Time Login process to the School Mess Payment System

[**user not valid**]

**[user valid]**

**[is staff] [is not staff]**

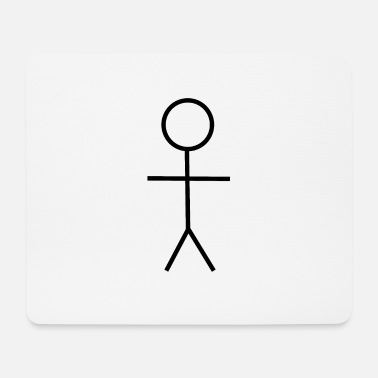
**Interaction Models: {**Use Case**}**



Login / Register

Place order & pay

Student



View pending orders

Update menu with available foods

Alter order list accordingly

Kitchen staff

The first Use case Diagram describes the relationship of a student with the system. The student can only register or login to the system in order to view the food menu. The student can also place an order through the system and pay for the order.

The second use case details interactions between kitchen staff and the system. Similarly kitchen staff also have to register or login to the system to access their respective functionalities which include: updating the food menu and viewing orders list and similarly updating the list once orders are fulfilled.

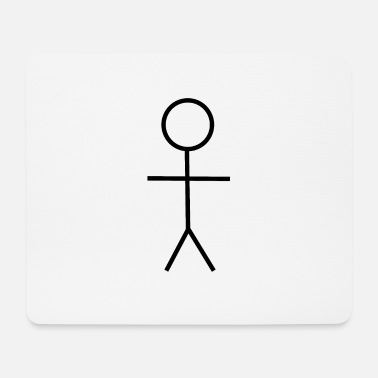
**Interaction models {**Sequence diagrams**}**

S: SMPS

U: user database

User authentication

Give user access



Login request

(Unique User ID)

Display food menu

Place order and pay

Student

**Structural Model: {**class diagrams**}**

1

1

User records

student

staff

orders

Food menu

1. 1

1 1

1

1..\*

1

1

1

1

1

1

|  |
| --- |
| **USER** |
| User#  National Id# |
| Register()  Login() |

|  |
| --- |
| **STUDENT** |
| Student ID |
| Getfoodmenu() |

|  |
| --- |
| **STAFF** |
| Staff ID |
| Getorderlist() |

**Behavioral Model:**

Student Login page

Validate user information

Food menu

Pick food items from menu

Checkout page

Pay for order

payment process

Payment validation

Receipt generation

Order forwarded to kitchen staff