

**Royal University of Phnom Penh**

**Department of Information Technology**

**Object-Oriented Analysis, Design and Programming**

**Prof. Dan Vournpheng**

**MiniMart POS System**

**Class: M1, Year 4**

**Group members**

1. Nou Sophavattey
2. So Visalsing
3. Tan Techsean
4. Tip Chhunlay

Academic Year: 2023-2024

# អារម្ភកថា

Minimart POS System គឺជាសម្រង់ឯកសារ ដែលបានចងក្រងឡើងដោយក្រុមនិស្សិតជំនាញវិទ្យាសាស្រ្តកុំព្យូទ័រ ថ្នាក់បរិញ្ញាបត្រឆ្នាំទី៤ ឆមាសទី២ នៃសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ ក្រោមការណែនាំរបស់ សាស្រ្តាចារ្យ ដាន វួនផេង ក្នុងគោលបំណងបង្កើតនូវប្រព័ន្ធមួយសម្រាប់គ្រប់គ្រងការលក់ទំនិញនៅក្នុងម៉ាត ដើម្បីឲ្យមានភាពងាយស្រួលដល់ម្ចាស់អាជីវកម្មក្នុងការគ្រប់គ្រងទិន្នន័យនៃការលក់ដូររបស់ពួកគាត់។

នៅក្នុងឯកសារនេះ ពួកយើងបានរៀបរាប់យ៉ាងលម្អិតអំពីដំណើរក្នុងការធ្វើប្រព័ន្ធគ្រប់គ្រងការលក់ទំនិញមួយនេះ។ ពួកយើងសង្ឃឹមថា ឯកសារមួយនេះអាចផ្តល់ជាអត្ថប្រយោជន៍ដល់អ្នកទាំងអស់គ្នា ជាពិសេស និស្សិតដែលកំពុងតែសិក្សាជំនាញវិទ្យាសាស្ត្រកុំព្យូទ័រ។

# សេចក្ដីថ្លែងអំណរគុណ

ពួកយើងសូមសម្តែងការដឹងគុណចំពោះ លោកគ្រូសាស្ត្រាចារ្យ ដាន វួនផេង ដែលបានលះបង់ចំណាយពេលវេលា កម្លាំងកាយ និងកម្លាំងចិត្តក្នុងការបង្រៀនផ្តល់ចំណេះដឹង ចែករំលែកបទពិសោធន៍ល្អៗ និងជួយណែនាំ បង្ហាញផ្លូវក្នុងការសិក្សាស្រាវជ្រាវ Project មួយនេះ។

ពួកយើងក៏សូមថ្លែងអំណរគុណយ៉ាងជ្រាលជ្រៅដល់ សាកលវិទ្យាធិការ សាកលវិទ្យាធិការរង សាស្រ្តាចារ្យ លោកគ្រូអ្នកគ្រូ គ្រប់ៗរូបដែលបានយកចិត្តទុកដាក់ ក្នុងការបង្រៀន និងចែករំលែកនូវចំណេះដឹងល្អៗ ដល់សិស្ស និស្សិត ឲ្យមានចំណេះជំនាញអាចយកទៅប្រើប្រាស់នាថ្ងៃអនាគត។

**មាតិកា**

អារម្ភកថា

សេចក្ដីថ្លែងអំណរគុណ

[1. គោលបំណង 1](#_Toc162995653)

[2. ការងាររបស់ Forms នីមួយៗ 2](#_Toc162995654)

[2.1. Splash Form 2](#_Toc162995655)

[2.2. Login Form 2](#_Toc162995656)

[2.3. Manage Category Form 3](#_Toc162995657)

[2.4. Manage Product Form 4](#_Toc162995658)

[2.5. Manage Seller​ Form 4](#_Toc162995659)

[2.6. Selling Form 6](#_Toc162995660)

[3. បង្ហាញចំណុចដែលប្រើប្រាស់ Design Patterns 7](#_Toc162995661)

[3.1. Observer Design Pattern 7](#_Toc162995662)

[3.2. Singleton Design Patten 8](#_Toc162995663)

[3.3. State Design Pattern 9](#_Toc162995664)

[3.4. Template Method Design Pattern 10](#_Toc162995665)

[3.5. Command Design Pattern 10](#_Toc162995666)

[សេចក្ដីសន្និដ្ឋាន 12](#_Toc162995667)

# គោលបំណង

នាពេលបច្ចុប្បន្ននេះផ្នែកបច្ចេកវិទ្យា Digital កាន់តែរីកចម្រើ និងដើរតួនាទីយ៉ាងសំខាន់ដែលតម្រូវអោយប្រើប្រាស់សឹងតែទៅលើគ្រប់វិស័យទាំងអស់ហើយក្នុងនោះក៏ចាំបាច់នៅក្នុងការធ្វើអាជីវកម្មផងដែរ។​ អាស្រ័យហេតុនេះហើយទើបក្រុមពួកខ្ញុំដែលជានិស្សិតនៅសាកសវិទ្យាល័យភូមិន្ទភ្នំពេញបានរួមគ្នាបង្កើត ប្រព័ន្ធគ្រប់គ្រងការលក់ក្នុងហាងទំនិញ(Minimart POS System) ដែលមានគោលបំណងដើម្បី៖

**ចំពោះ Admin**

* គ្រប់គ្រងទៅលើ Product ដែលអាចធ្វើការ Add, update, និង Delete information របស់ Product នីមួយៗ
* គ្រប់គ្រងទៅលើ Category ដែលអាចធ្វើការ Add, Update, និង Delete information របស់ Category ទៅតាមឈ្មោះទំនិញនីមួយៗ
* គ្រប់គ្រងទៅលើ Seller ដែលអាចធ្វើការ Add, Update, និង Delete information របស់ Seller ម្នាក់ៗ

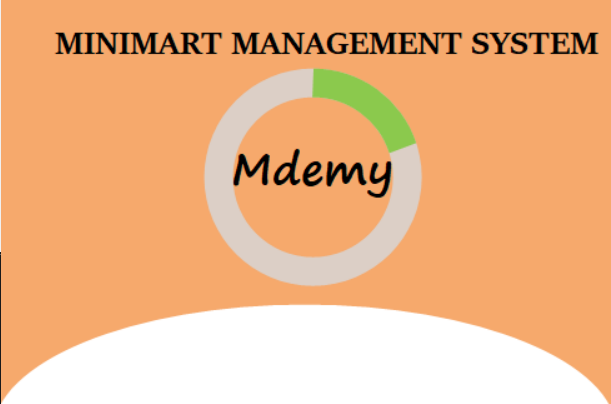
**ចំពោះ Seller**

* គ្រប់គ្រងទៅលើការលក់ឬ Selling ដែលអាចធ្វើការ Add Orders, Remove Orders, Clear Orders, Add Bill, និង Print Bill

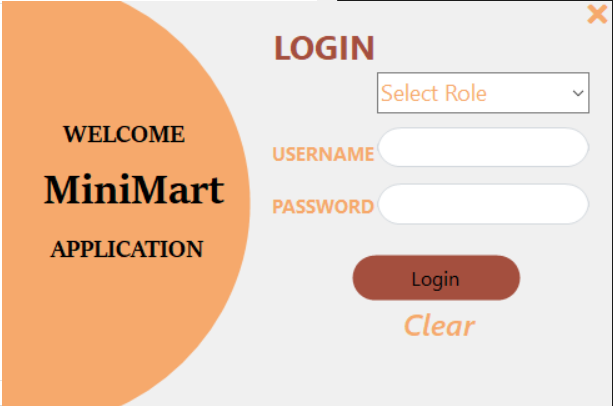
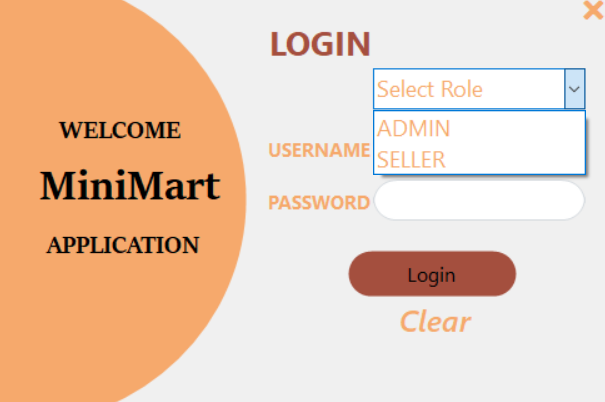
ដូចនេះនៅក្នុង Project Report មួយនេះ ក្រុមយើងខ្ញុំនឹងបង្ហាញពីការប្រើប្រាស់និងពន្យល់អំពី Forms នីមួយៗដែលមាននៅក្នុង Project មួយនេះ និង Design Patterns មួយចំនួនក្នុងកម្មវិធីប្រព័ន្ធគ្រប់គ្រង​ការលក់ក្នុងហាងទំនិញ(Minimart POS System) ដោយប្រើភាសា C# នៅក្នុងគម្រោងកម្មវិធី Windows Visual Studio 2019 ។

# ការងាររបស់ Forms នីមួយៗ

## Splash Form

វាគឺជា Form ដែលយើងឃើញដំបូងគេ នៅពេលចូលទៅកាន់កម្មវិធី។ វាបង្ហាញនូវការ loading ដើម្បីចូលទៅក្នុងកម្មវិធីនេះ។

## Login Form

Login Form គឺជា Form ដែលតម្រូវឲ្យ user ជ្រើសរើស role ហើយបញ្ជូល username និង password ដើម្បី log in ចូលប្រើប្រាស់កម្មវិធី។

* Field Select Role: តម្រូវឲ្យ user ជ្រើសរើស role តើគាត់ជា admin ឬ seller។
* Field Username: តម្រូវឲ្យ user បញ្ចូល username របស់គាត់។
* Field Password: តម្រូវឲ្យ user បញ្ចូល password របស់គាត់។
* Button Login: នៅពេល user ចុច Button Login ប្រព័ន្ធនឹងផ្ទៀងផ្ទាត់ថាតើ username និងpassword ដែលបានបញ្ចូលត្រឹមត្រូវដែរឬទេ បើត្រឹមត្រូវ វានឹងនាំ user ទៅកាន់ Form បន្ទាប់។
* Button Clear: នៅពេល user ចុច Button Clear វានឹងលុប username និង password ដែលអ្នកប្រើប្រាស់បានបញ្ចូល។

## Manage Category Form

Category Form គឺជា Form ដែលអនុញ្ញាតឲ្យអ្នកប្រើប្រាស់អាចគ្រប់គ្រង Categories បាន តាមរយៈការ add (id, name, description) , update ពត៌មានទាំងអស់នោះ និង delete categories បាន។ នៅក្នុង Form នេះ មានButtonចំនួន៣គឺ៖

* Button Add: នៅពេលអ្នកប្រើប្រាស់ចុចលើប៊ូតង់នេះ វានឹងបញ្ចូល id, name និង description របស់ category ថ្មី។
* Button Update: វាអនុញ្ញាតឲ្យអ្នកប្រើប្រាស់ update នូវពត៌មានរបស់ category ណាមួយ។
* Button Delete: វាអនុញ្ញាតឲ្យអ្នកប្រើប្រាស់ delete នូវពត៌មានរបស់ category ណាមួយ។

## Manage Product Form

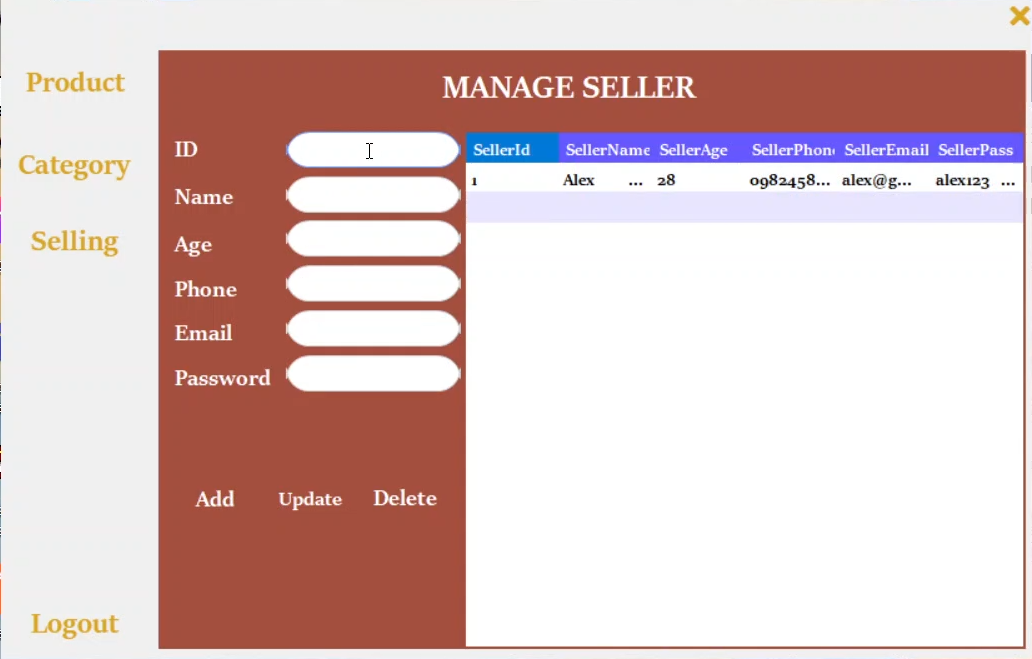
Product Form គឺជា Form ដែលអនុញ្ញាតឲ្យអ្នកប្រើប្រាស់អាចគ្រប់គ្រង product បាន តាមរយៈការ add (id, name, price, quantity, category), update ពត៌មានទាំងអស់នោះ និង delete product បាន។ លើសពីនេះ អ្នកប្រើប្រាស់អាចចូលមើល products ទៅតាម category នីមួយៗបាន។ នៅក្នុង Form នេះ មានButtonចំនួន៣គឺ៖

* Button Add: នៅពេលអ្នកប្រើប្រាស់ចុចលើប៊ូតង់នេះ វានឹងបញ្ចូល id, name, price, quantity និង category របស់ product ថ្មី។
* Button Update: វាអនុញ្ញាតឲ្យអ្នកប្រើប្រាស់ update នូវពត៌មានរបស់ product ណាមួយ។
* Button Delete: វាអនុញ្ញាតឲ្យអ្នកប្រើប្រាស់ delete នូវពត៌មានរបស់ product ណាមួយ។

## Manage Seller​ Form

នៅក្នុង Form នេះគឺអនុញ្ញាតិឲ្យAdminគាត់ធ្វើការគ្រប់គ្រង់ទៅលើបុគ្គលិករបស់គាត់ដែលមានដូចជា បន្ថែមបុគ្គលិក​ និងUpdate ព័ត៌មាន របស់​បុគ្គលិក និងធ្វើការលុបចោលជាដើម។

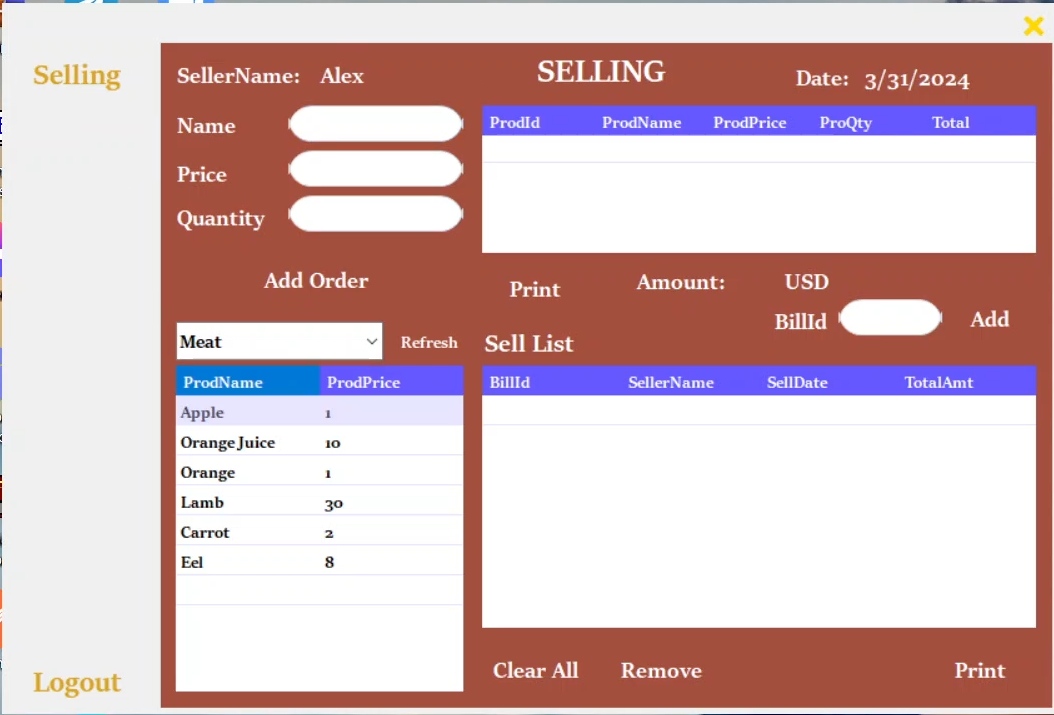
ដើម្បីធ្វើការបង្កើតនូវបុគ្គលិកថ្មីបាន ជាដំបូងតម្រូវឲបញ្ចូលនូវទិន្នន័យមួយចំនួនជាមនុសសិន៖

* **Field ID:** សម្រាប់បញ្ចូលនូវ ID របស់បុគ្គលិក
* **Field Name:** សម្រាប់បញ្ចូលនូវ ឈ្មោះរបស់បុគ្គលិក
* **Field Age:** សម្រាប់បញ្ចូលនូវអាយុ​របស់បុគ្គលិក
* **Field Email:** សម្រាប់បញ្ចូលនូវ Email របស់បុគ្គលិក
* **Field Phone:** សម្រាប់បញ្ចូលនូវលេខទូរស័ព្ទ របស់បុគ្គលិក
* **Field Password:** សម្រាប់បញ្ចូលនូវលេខសំងាត់របស់បុគ្គលិក

ហើយនៅក្នុង Form នេះដែរគឺប្រើប្រាស់នូវ Button ចំនួន៣គឺ៖

* ADD: ប្រើសម្រាប់បន្ថែមបុគ្គលិកថ្មី
* UPDATE: ប្រើសម្រាប់កែប្រែព័ត៌មាន ឬទិន្នន័យរបស់បុគ្គលិកដែលមានស្រាប់
* DELETE: ប្រើសម្រាប់់លុបចោល នូវបុគ្គលិកណាមួយដែលមិនត្រូវការប្រើ ឬឈប់ប្រើប្រាស់

## Selling Form

 នៅក្នុង Form នេះដែរគឺអនុញ្ញាតិឲអ្នកលក់គាត់ធ្វើការគ្រប់គ្រងទៅលើការលក់ ដែលមានដូចជា ជ្រើសរើសផលិតផល និងព្រីនចេញជាវិក្ក័យបត្រជាដើម។

ចំពោះ Form នេះដែរគឺយើងបានប្រើប្រាស់នូវ Tables ចំនួន៣គឺ

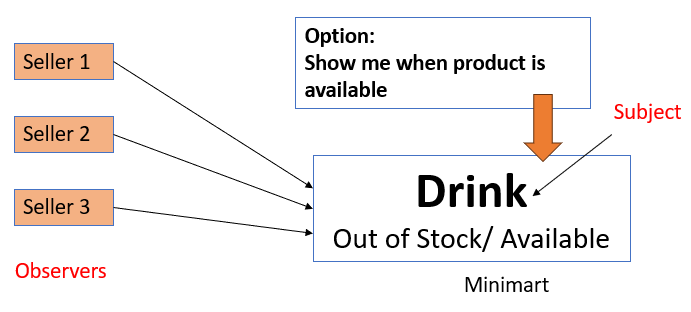
* Products Table: ប្រើសម្រាប់ឲអ្នកលក់ធ្វើការជ្រើសរើសនូវ Products ណាដែល customers ត្រូវការ បន្ទាប់មកទៀតវានឹងធ្វើការបង្ហាញទិន្នន័យរបស់ផលិតដែលបានជ្រើសរើសនោះទៅកាន់ Text boxs ខាងលើដែលមានដូចជា Name, Price, Quantity ដែលអាចដាក់ចំនួនទៅតាមការចង់បានរបស់អតិថិជន
* Orders Table: ប្រើសម្រាប់បង្ហាញនូវរាល់ទំនិញណាដែលបានជ្រើសរើសរួចឲ្យចូលទៅកាន់ Orders Table ក្រោយពីការចុចលើ ADD Button
* Bill Table: ប្រើសម្រាប់បង្ហាញនូវចំនួន orders ដែលបានលក់ចេញ ដែលវាធ្វើការបង្ហាញនូវទិន្ន័យ BillID, អ្នកលក់ កាលបរិច្ឆេត និងចំនួនសរុបនៃ orders នីមួយ

នៅក្នុង Form នេះទៀតសោត គឺមានដាក់ឲប្រើប្រាស់នូវ Button ចំនួន៥៖

* Add Order: ប្រើសម្រាប់បញ្ចូលផលិតផល់ដែលបានជ្រើសរើសរួចចូលទៅកាន់ Orders Table
* Add Bill: ប្រើសម្រាប់កត់បញ្ចូលនូវ Orders ទៅជាវិក្ក័យបត្រ ដើម្បីគ្រប់គ្រងមើលថាតើ ក្នុងថ្ងៃនេះលក់បានប៉ុន្មាន orders
* Clear All: ប្រើសម្រាប់លុបនូវរាល់ orders ទាំងអស់ចោល
* Remove: ប្រើសម្រាប់លុបចោលតែ order or bill ដែលបានជ្រើសរើសរួចប៉ុណ្ណោះ
* Print: ប្រើសម្រាប់ព្រីន orders or bill ចេញជា hardcopy សម្រាប់ជូនទៅកាន់អតិថិជន

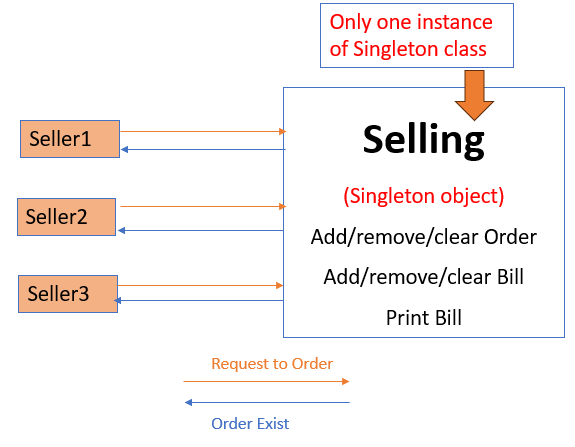
# បង្ហាញចំណុចដែលប្រើប្រាស់ Design Patterns

## Observer Design Pattern

ចំណុចដែលបង្ហាញពីការប្រើប្រាស់ Observer Pattern គឺនៅពេលដែល Seller ធ្វើការ Add Order។ មុនពេលធ្វើការ Add order ទៅលើ product​ ណាមួយ Seller ត្រូវ check មើលឬ filter នៅក្នុង Table Category ជាមុនសិន។ ឧទាហរណ៍នៅពេលដែល Drink អស់ពីស្តុកឬ out of stock នោះវាមិនបង្ហាញចេញនៅក្នុង Table Category នោះទេ។ ប៉ុន្តែប្រសិនបើប៉ុន្មានថ្ងៃក្រោយ Admin ធ្វើការ Add ស្តុកឡើងវិញនោះ Sellers នឹងអាចឃើងនៅក្នុង Table Category បាន។

ដូចនេះហើយអ្វីដែល Minimart POS ធ្វើគឺវាបង្ហាញដល់ Sellers ថាតើ product ណានៅក្នុង stock និង product ណាដែល out of stock។ ដូចដែលយើងដឹងរួចហើយថា Observer Design Pattern មាន components ចំនួន ២ គឺ Observer និង Subject ដែល (Seller1, Seller2, and Seller3) ជា Observers ហើយ Drink គឺជា Subject ។

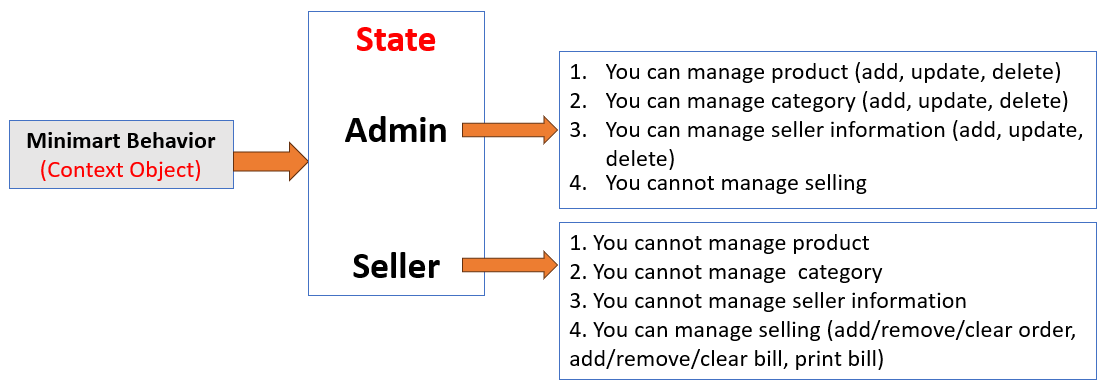
## Singleton Design Patten

****នៅក្នុង Minimart POS System នេះ ចំណុចដែលបង្ហាញពីការប្រើប្រាស់ Singleton Design Patten គឺពាក់ពន្ធ័នឹងការលក់ឬ Selling។ នៅពេលដែល Sellers ធ្វើការ Loggin នោះរាល់ Sellers ទាំងអស់នឹងអាចទាយយកទិន្នន័យពី Products និង Categories បានដូចគ្នាៗ ដោយសារតែព័ត៌មានទាំងនោះត្រូវបាន stored ក្នុង Database តែមួយ។ ឧទាហរណ៍ថាបើ Seller1 loggin ចូលក្នុង system ហើយ seller1 នោះអាចធ្វើការ Add/remove/clear order, Add/remove/clear bill, និង Print bill បាន។ ដែលនេះមានន័យថា Seller2, Seller3,…SellerN ​ក៏អាចធ្វើ process ទាំងនោះបានដូចគ្នា។ Singleton Design Patten ត្រូវបានគេប្រើជាញឹកញាប់សម្រាប់ Loggin, Database connections, Caching, Thread pools, និង Data Sharing។

## State Design Pattern

Minimart POS behavior គឺជាចំណុចដែលបង្ហាញពីការប្រើប្រាស់ State Design Pattern នៅក្នុង Project នេះ។ ឥឡូវនេះយើងក្រឡេកមកមើល State Design Pattern Diagram នៅក្រោមនេះដោយចាត់ទុកថា internal state របស់ Minimart POS System Behavior (Context Object) គឺជា Admin​​ មានន័យថានៅពេលដែល Admin ធ្វើការ Loggin ចូល System នោះ Admin អាចធ្វើការទៅលើ operations ដូចជា៖

1. Admin អាចធ្វើការ manage ទៅលើ product information ដូចជា add, update, និង delete
2. Admin អាចធ្វើការ manage ទៅលើ category ដូចជា add, update, និង delete
3. Admin អាចធ្វើការ manage ទៅលើ seller information ដូចជា add, update, និង delete
4. Admin មិនអាចធ្វើការ manage ទៅលើ selling បាននោះទេ

****

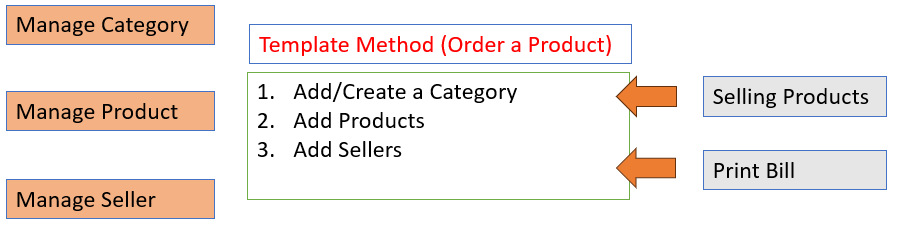
ប្រសិនបើ Seller ជាអ្នកធ្វើការ Loggin ចូល System វិញនោះ​ internal state របស់ Minimart POS System Behavior (Context Object) នឹងធ្វើការផ្លាស់ប្ដូរទៅជា Seller។ ដូចនេះ operations ដែល Seller អាចធ្វើការទៅលើបានមានដូចជា៖

1. Seller មិនអាចធ្វើការ manage ទៅលើ product information បាននោះទេ
2. Seller មិនអាចធ្វើការ manage ទៅលើ category បាននោះទេ
3. Seller មិនអាចធ្វើការ manage ទៅលើ seller information បាននោះទេ
4. Seller អាចធ្វើការ manage ទៅលើ selling ដូចជា add/remove/clear order, add/remove/clear bill, និង print bill

យើងត្រូវចំណាំថា ប្រសិនបើ state គឺជា Admin នោះអាចធ្វើការទៅលើ operations មួយចំនួននៅពេល Loggin ចូល System ប៉ុន្តែប្រសិនបើ Seller ជាអ្នកធ្វើការ Loggin ចូល System វិញនោះ Seller គឺជា state ដែលអាចធ្វើការទៅលើ operations មួយចំនួនបានដូចគ្នា។ ដូចនេះហើយ internal state គឺអាស្រ័យការផ្លាស់ប្ដូររបស់ Minimart POS System Behavior (Context Object)។

## Template Method Design Pattern

ចំពោះការប្រើប្រាស់ Template Method Design Pattern ត្រូវបានបង្ហាញនៅពេលដែលយើងចង់ធ្វើការលើ Selling Product និង Print Billing ដូចនេះមុនពេលដំណើរការមុខងារទាំងនេះបាន យើងត្រូវឆ្លងកាត់ដំណាក់កាលមួយចំនួនសិន។ ឧទាហរណ៍ថា យើងចង់ធ្វើការ Selling Products ដូចនេះជាដំបូង Admin ត្រូធ្វើការ Add ឬ Create Category ជាមុនសិន បន្ទាប់មក Admin ត្រូវ Add Products ចូលទៅក្នុង Stock និងចុងក្រោយធ្វើការ Add Sellers ដើម្បីឱ្យ Sellers ដំណើរការទៅលើ Selling Products និងដើម្បីអាចធ្វើការ Print Bill បានផងដែរ។

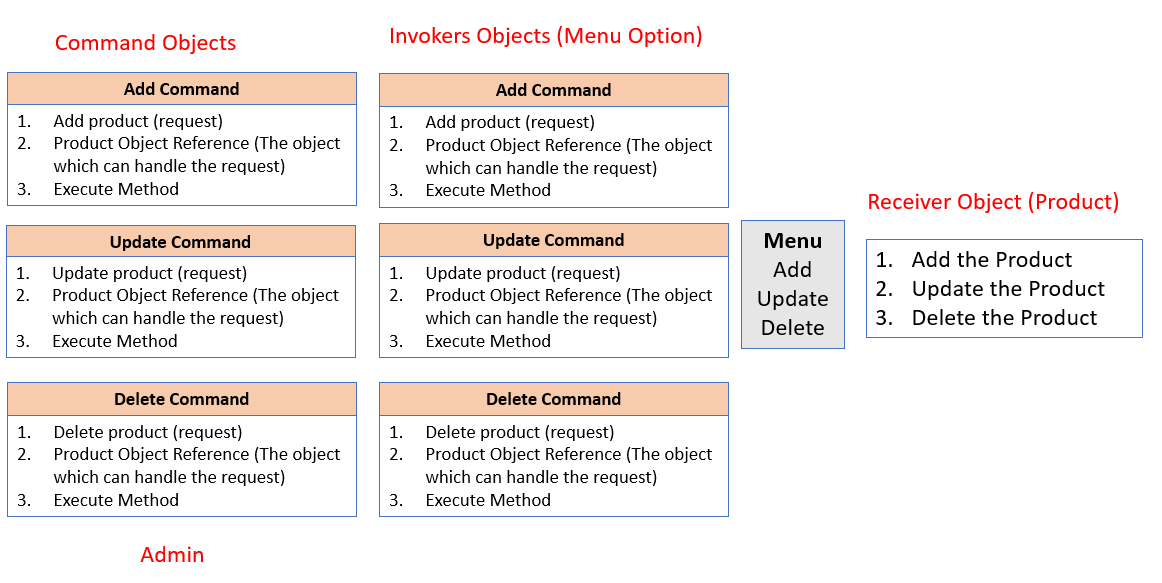
****

## Command Design Pattern

ចំណុចដែលបង្ហាញពីការប្រើប្រាស់ Command Design Pattern គឺនៅក្នុងការ Manage Product។ ដោយយើងបានឃើញថានៅក្នុង Diagram ខាងក្រោមមាន Command ចំនួន 3 ដូចជា Add, Update, និង Delete។

* Command ទីមួយគឺ Add Command ដែលវា request ដើម្បីធ្វើការ Add Product និងមាន Receiver Object ជា Product Object ហើយ The Execute method នឹង call Add method នៃ Product object។
* Command ទីពីគឺ Update Command ដែលវា request ដើម្បីធ្វើការ Update Product និងមាន Receiver Object ជា Product Object ហើយ The Execute method នឹង call Update method នៃ Product object។
* Command ទីបីគឺ Delete Command ដែលវា request ដើម្បីធ្វើការ Delete Product និងមាន Receiver Object ជា Product Object ហើយ The Execute method នឹង call Delete method នៃ Product object។

The invoker object (i.e. Menu options) មិនដឹងពីរបៀប handle the request នោះទេ អ្វីដែល invoker object នឹងធ្វើការគឺ call Execute method នៃ command object ដូចនេះហើយនៅពេលដែល Admin ធ្វើការ clicks ទៅលើ Add menu នោះវានឹងធ្វើការ execute ទៅលើ Execute method នៃ Add command នោះ ឬវានឹងធ្វើការ Add ចូលទៅក្នុង Table។ ដូចគ្នានេះដែល ប្រសិនបើ Admin ធ្វើការ clicks ទៅលើ Delete menu នោះវានឹងធ្វើការ execute ទៅលើ Execute method នៃ Delete command នោះ ឬវានឹងធ្វើការ Delete ចេញពីក្នុង Table។



# សេចក្ដីសន្និដ្ឋាន

បន្ទាប់ពីបានធ្វើ Project នៅក្នុងមុខវិជ្ជា Object-Oriented Analysis, Design and Programmingអំពី MiniMart POS System ខាងលើនេះរួចមក ក្រុមយើងខ្ញុំឃើញថា Project នេះបានផ្តល់នូវចំណេះដឹង និងឱកាសជាច្រើនដល់ពួកខ្ញុំជាច្រើនក្នុងការអនុវត្តន៍នូវអ្វីដែលបានរៀននៅក្នុងថ្នាក់ដូចជា OOP in C# language, design form, និង design Patterns មួយចំនួនផងដែរ។ ក្រុមយើងខ្ញុំសង្ឃឹមថា project មួយនេះអាចជាប្រយោជន៍ដល់ការប្រកបអាជីវកម្មផងដែរ។

ដូចនេះក្នុងនាមយើងជាកូនខ្មែរ និងជាអ្នកសិក្សាខាងផ្នែកបច្ចេកវិទ្យាយើងត្រូវចូលរួមអភិវឌ្ឍន៍លើគ្រប់វិស័យជាពិសេសផ្នែកបច្ចេកវិទ្យាអោយមានកាន់តែមានភាពរីកចម្រើន ដើម្បីទុកជាប្រយោជន៍ដល់អ្នកសិក្សាជំនាន់ក្រោយៗអោយបានស្វែងយល់។