**ШИНЖЛЭХ УХААН ТЕХНОЛОГИЙН ИХ СУРГУУЛЬ**

**Мэдээлэл холбооны технологийн сургууль**

**A blue and yellow sign with a gear and drops

Description automatically generated**

**БИЕ ДААЛТЫН АЖЛЫН**

**ТАЙЛАН**

Программчлалын дадлага

|  |  |  |  |
| --- | --- | --- | --- |
| **Бие даалтын нэр:** |  | Төслийн хөгжүүлэлт | |
| **Хичээл заасан багш:** |  | **А.Отгонбаяр** | |
| **Бие даалт гүйцэтгэсэн:** | Оюутан 1: | И.Эрдэнэбаяр | B232270136 |
|  | Оюутан 2: | Ч.Мөнхбаяр | B242270045 |
|  | Оюутан 3: | Б. Цогжавхлан | B222270838 |
|  | Оюутан 4: | А. Базарсад | B242270065 |

Гарчиг

[ЗУРГЫН ЖАГСААЛТ 2](#_Toc200409853)

[KASS JAVA POS SYSTEM BACKEND 3](#_Toc200409854)

[Танилцуулга: 3](#_Toc200409855)

[Шаардлагын тодорхойлолт: 3](#_Toc200409856)

[Кодын бүтэц, загвар: 4](#_Toc200409857)

# ЗУРГЫН ЖАГСААЛТ

[Зураг 1(uml (umldraw, n.d.)) 4](#_Toc200409797)

# KASS JAVA POS SYSTEM BACKEND

## Танилцуулга:

Энэхүү мини төсөл нь Java хэл дээр бичигдсэн, ресторан болон кафе зэрэг үйлчилгээний газарт зориулсан POS системийн backend шийдэл юм. Төслийн гол зорилго бол хэрэглэгчийн захиалга, ширээ, бүтээгдэхүүн болон төлбөрийн мэдээллийг бүртгэж, боловсруулж, системийн гол логикийг сервер талд хэрэгжүүлэх явдал байлаа.

Энэ backend систем нь консолоос ажиллах CLI интерфэйстэй бөгөөд Log4j2 ашиглан лог хөтлөлт хийдэг. Бүх үндсэн CRUD үйлдлүүдийг хийдэг бөгөөд хэрэглэгчтэй интерактив харилцаа үүсгэдэг.

Онцлогууд:

* Категори болон бүтээгдэхүүнийг удирдах (CategoryRepository, ProductRepository)
* Захиалга бүртгэх, хадгалах (OrderService)
* Төлбөрийн тооцоолол хийх (CheckoutService)
* Ширээний менежмент (TableRepository, TableRep)
* Exception болон лог менежмент (Log4j2 ашигласан)
* CLI орчноос удирдах боломж (MainCLI)
* JDBC ашиглан өгөгдлийн сантай холбогдсон бүтэцтэй

## Шаардлагын тодорхойлолт:

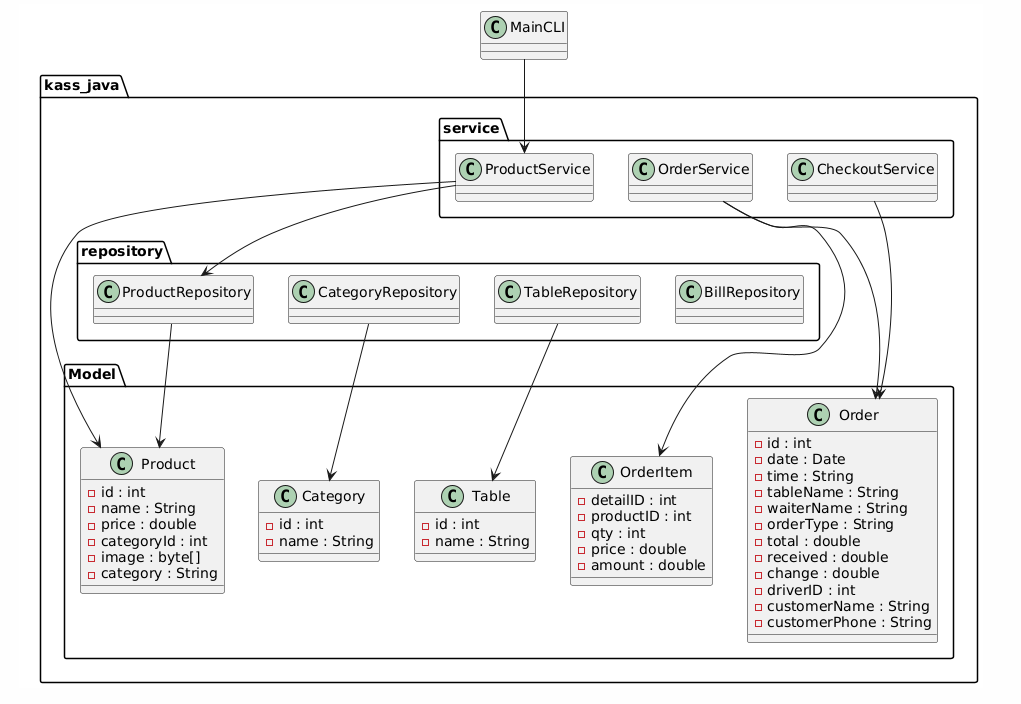
Төслийн зорилго бол POS системийн гол хэсэг болох backend-ийг боловсруулах. Систем нь:

* Шинэ бүтээгдэхүүн, категори нэмэх, устгах, шинэчлэх;
* Захиалга, ширээ бүртгэх, бүртгэлүүдийг удирдах;
* Төлбөрийн процесс хийх, хариулт мөнгө тооцоолох;
* CLI орчноор дамжуулан бүх өгөгдлийг удирдах;
* Лог бичих (console + файл руу);
* Бүтээгдэхүүн шалгах, үлдэгдэл хянах боломжтой байх ёстой.

## Кодын бүтэц, загвар:

**Model:** Product, Category, Order, OrderItem, Table, Bill  
**Repository:** CategoryRepository, ProductRepository, TableRepository, BillRepository  
**Service:** ProductService, OrderService, CheckoutService, StockManagementService  
**Entry Point:** MainCLI

UML диаграм (pptx дээр хавсаргах):



Зураг 1(uml (umldraw, n.d.))

**Сурсан зүйл, бэрхшээл, ирээдүйн сайжруулалт**

* **Сурсан зүйлс:** Java backend-ийн архитектур, Log4j logging, CLI менежмент, exception зохицуулалт, Maven төслийн менежмент, JUnit тест.
* **Бэрхшээлүүд:** Log4j config detection, CLI дээр NumberFormatException гарах асуудлууд, test runner-уудын тохиргоо.
* **Сайжруулалт:** GUI интерфейс нэмж хөгжүүлэх, database persistence бүрэн холбох (JDBC), хэрэглэгч бүрийн эрх удирдах модулиуд нэмэх, JUnit тестийг илүү өргөжүүлэх.

**Кодын баримтжуулалт**

* Бүх класс, функц дээр Javadoc тайлбар бичигдсэн.
* logger.info/warn/error() хэлбэрээр лог бичиж, логийн түвшин бүр тодорхой.
* Exception гарсан тохиолдолд лог болон системд ойлгомжтой мессеж өгдөг.
* Код нь нэг үүрэг гүйцэтгэхээр задарсан, Service, Repository гэсэн хандалтаар хуваагдсан.