**Los Pollos Hermanos**

**Návrhová dokumentace (Detailní design)**

Dokument vytvořen pro potřeby předmětu BI-SI1

Autoři: : Martin Horák, Tomáš Böhm, Markéta Kocourková, Matěj Formánek, Tomáš Krejčík, Ondřej Sakala, Patrik Cinert

**Obsah**

1. Návrh architektury 3

1.1 BL 5

1.1.1 worker 6

1.2 DL 6

1.2.1 dao 7

1.2.2 entity 8

1.3 PL 10

1.3.1 controller 10

1.3.2 view 10

2. Realizace případů užití 12

2.1 Realizace UC1 - Evidovat várku 12

2.1.1 Model komunikace 12

2.2 Realizace UCy - Login 13

2.2.1 Sequence Diagram 13

2.2.1.1 Actors 13

2.2.1.2 Components 13

2.3 Realizace UCx - Odvoz 13

2.3.1 Actors 14

2.3.2 Components 15

# Návrh architektury

Architektura aplikace je navržena jako třívrstvá. Rozhraní mezi jednotlivými vrstvami bude přesně definováno během implementace aplikace. Rozhraní umožní snadnou výměnu jednotlivých vrstev.

Persistence dat (ORM) = Peewee (<https://github.com/coleifer/peewee>)

Uživatelské rozhraní (GUI) = Streamlit (<https://streamlit.io/>)



Obrázek 1 - Návrh architektury



Obrázek 2 - Návrhový model tříd

## BL

Veškerá business logika, kterou je nutné ze zadání implementovat, bude obsažena v balíčcích v této vrstvě. Kromě této logiky bude vrstva obsahovat i definici rozhraní pro komunikaci s prezentační vrstvou. Třídy v této vrstvě budou organizovány do balíčků, dle business oblastí.



Obrázek 3 - BL

### worker

Balíček obsahuje třídy, které se starají o propojení databových prvků do prezenční vrstvy.



Obrázek 4 - Workers

## DL

Datová/technická vrstva aplikace, která bude obsahovat třídy umožňující persistentně ukládat /načítat objekty z datového úložiště.

Pro persistetní ukládání dat využíváme existující ORM framework - Peewee.



Obrázek 5 - DL

### dao

Tento balíček obsahuje definici rozhraní, kterou může prezentační vrstva využívat pro načítání/ukládání dat. Veškeré implementace těchto rozhraní jsou ukládány do samostatných balíčků vnořených do tohoto balíčku



Obrázek 6 - dao

### entity

Balíček obsahuje definici datových objektů, se kterými aplikace pracuje. Tento balíček využívají všechny balíčky, které potřebují s těmito daty pracovat včetně prezentační vrstvy.



Obrázek 7 - entity

## PL

Prezentační vrstva aplikace, která umožňuje zobrazování stavu aplikace uživateli a předávat uživatelské požadavky do business vrstvy. Tyto požadavky třídy v prezentační vrstvě transformují tak, aby odstranily závislost na použité technologii

(desktopová aplikace, http request).



Obrázek 8 - PL

### controller

Tento balíček slouží třídám, které zpracovávají uživatelské vstupy a transformují je na volání metod datové vrstvy.



Obrázek 9 - controler

### view



Obrázek 10 - view

# Realizace případů užití

## Realizace UC1 - Evidovat várku

### Model komunikace



Obrázek 11 - Evidovat várku

## Realizace UCy - Login

### Sequence Diagram

Sequence diagrams are used to model the logic of usage scenarios by showing the information passed between objects in the system through the execution of the scenario. *BABOK® Guide v3*



Obrázek 12 - Login diagram

#### Actors

#### Components

## Realizace UCx - Odvoz

Sequence diagrams are used to model the logic of usage scenarios by showing the information passed between objects in the system through the execution of the scenario. *BABOK® Guide v3*



Obrázek 13 - Evidence odvozů