|  |  |
| --- | --- |
| **Project – External Documentation** |  |
| BK01  Object Oriented Analysis & Design |
| <**Odd**/Even/Compact> Semester Year <2023/2024> |

* Project Title

Mystic Grills

* Introduction

Mystic Grills merupakan sebuah aplikasi manajemen restoran yang inovatif yang didesain untuk mempermudah operasional dan meningkatkan pengalaman makan secara menyeluruh. Dengan beragam fitur seperti manajemen pengguna, penyesuaian menu, dan sistem pemesanan yang efisien, Mystic Grills menyediakan platform yang lancar dan intuitif bagi staf restoran maupun pelanggan. Tujuan utama Mystic Grills adalah merevolusi manajemen restoran dengan memanfaatkan teknologi guna meningkatkan efisiensi, meningkatkan kepuasan pelanggan, dan mendukung kesuksesan restoran.

* Report / Documentation
  + Download requirment2 yang dibutuhin dari web bluejack antara lain:
    - Eclipse 2020.6 R (IDE yang saya gunakan)
    - Visual Paradigm Community Edition 17.1 (buat liat design requirement project)
    - MySQL Java Connection Library 8.0.24 (buat konekin ke database)
    - Java 11.0.18 (JDK java yang saya gunakan)
    - XAMPP 8.0.7 (Buat run db mysql local)
    - JavaFX 17.0.7 (library java fx yang saya gunakan)
  + Setting Configure Build Path pada project java (sql connection, javafx)

A screenshot of a computer

Description automatically generated

Untuk sql connection bisa langsung Add External JARs terus cari file mysql connectornya

Untuk javafx bisa Add Library dulu, lalu pilih user library -> user libraries -> new -> input nama -> Add External JARs -> cari file javafx17 -> ke folder lib, pilih semua file jarnya -> masukan path dari zip yang beranama src pada folder java fx ke dalam setiap file jar pada bagian source attachment, and ready sudah.

* + Untuk run project kita harus memasukan vm argument terlebih dahulu

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* + Bikin Project
    - Structur File (MVC)

A screenshot of a computer

Description automatically generated

Project dibuat dengan menggunakan MVC. Package model disini berisi class2 untuk model yang merepresentasikan table database. Package model.dao berisikan class yang berguna untuk connection db dan juga logic-logic yang berhubungan dengan CRUD data ke database. Package controller dan view sesuai dengan fungsi pada umumnya, controller untuk perantara model dan view, dan view untuk tampilan user. Untuk view saya pisah menjadi beberapa package sesuai dengan view yang dibutuhkan oleh role tertentu. Untuk package view itu berisikan class2 view yang tidak terkait dengan role seperti login, register, dan juga template untuk membuat class view snantinya.

* + - View

A screenshot of a computer program

Description automatically generated

Sedikit penjelasan logika codingan view, package view menyimpan template untuk membangun page-page kedepannya. TemplateView.java digunakan untuk membuat header dan menentukan header yang sesuai untuk role yang login, dan juga membuat footer. BaseView dan BaseViewDetail akan mengextends TemplateView untuk mendapatkan header dan footer yang sudah siap pakai. Dalam BaseView dan BaseViewDetail ini dibuatlah template layout scene yang akan digunakan oleh class-class view yang akan dibuat. Class view akan memilki template seperti header dilanjut content dilanjut footer. BaseView dan BaseViewDetail merupakan abstract class yang nantinya diturunkan ke setiap class view, kedua class ini memiliki abstract method content() yang nantinya harus di override oleh setiap class view, diisi dengan content yang ingin ditampilkan. Pada akhirnya setiap class view akan memilki method display() yang merupakan method turunan dari class BaseView/BaseViewDetail yang berisikan header, content, dan juga footer, method display() inilah yang akan dipanggil dan menjadi tampilan user. Note: Class BaseView akan diturunkan ke majority of view class, sedangkan BaseViewDetail akan diturunkan ke class yang bertujuan untuk menampilkan detail dari suatu hal seperti order.

* + - Tambahan Package Factory

A close up of words

Description automatically generated

Package Factory ini sederhananya saya gunakan untuk membuat header. Alasan saya membuat class factory terpisah seperti ini karena biar rapih codenya sehingga lebih mudah dibaca. Jadi class TemplateView hanya perlu memanggil method header yang sesuai dengan role yang login dari headerFactory.

A screenshot of a computer code

Description automatically generated

* + - SessionManager Class

A screenshot of a computer program

Description automatically generated

SessionManager Class disini saya gunakan untuk menyimpan user yang telah berhasil login dan sedang menggunakan aplikasi, selain itu SessionManager ini juga berguna untuk mengetahui user sedang berada dipage view yang mana, SessionManager akan menyimpan id dari label yang diclick user saat pindah page pada header. Label yang memiliki id yang sesuai dengan id label yang tersimpan pada class session manager maka akan memiliki style khusus yang berguna sebagai penanda current page.

* + - Tambahan class model dan controller untuk OrderForTableView

OrderForTableView disini saya buat karena saya ingin menampilkan data pada TableView yang dimana data tersebut merupakan campuran dari 2 table yaitu MenuItem dan OrderItem. MenuItem untuk kolom id, nama, price sedangkan OrderItem untuk mendapat kolom quantity. Pada TableView hanya bisa menerima 1 sumber data oleh karena itu saya membuat sebuah class baru OrderForTabelView yang memiliki attributes dari kolom2 yang dibutuhkan tadi (menuItemId, menuItemName, menuItemPrice, quantity), object dari class ini lah yang akan menjadi sumber dari TableView tersebut.

Controller untuk OrderForTabelView akan memiliki sebuah ObservableList< OrderForTableView> local yang akan digunakan untuk menyimpan data sementara sebelum dimasukan ke dalam database. Jadi perubahan yang dilakukan user seperti menambah item, update item, remove item, selama user belum melakukan submit maka perubahan tersebut akan disimpan pada ObservableList< OrderForTableView> pada controller ini. Untuk coding detailnya dapat dibaca pada file java yang saya kumpulkan.

* + - Code selebihnya

Untuk code selebihnya seharusnya sudah sesuai dengan requirement pada diagram, hanya saja tentunya ada penambahan yang saya lakukan seperti pada model dan controllernya. Penambahan ini biasanya berupa method yang dapat memudahkan saya dalam memanipulasi data.

* + - Penjelasan tampilan
* Login

A login form with text and a log in box

Description automatically generated

Login page tidak ada yang special hanya validasi seperti biasa sesuai yang diminta soal, tidak boleh empty dan jika sesuai dengan data di database maka berhasil login.

* Register

A screenshot of a login form

Description automatically generated

Register Page tidak ada yang special hanya validasi seperti biasa sesuai yang diminta soal, tidak boleh empty, email harus unique, password minim 6 karakter, confirm password match, role default customer, jika semua sesuai maka berhasil register dan user akan mendapat pesan. Jika user ingin login maka user harus balik ke login page melalui label login yang berwarna biru.

* All Role - Home page (pada gambar con dari Admin Home)

A screenshot of a computer

Description automatically generated

Tampilan home page setelah user berhasil login, tampilan ini sebenarnya tidak ada pada soal namun saya sudah terlanjut buat home untuk semua role, pada akhirnya home page untuk setiap role akan berisika greetings aja dan ada tulisan have a good day (username yang login). Logout pada header berguna untuk mengeluarkan user dan balik ke login page.

* Admin – View Users

A screenshot of a computer

Description automatically generated

Admin bisa pilih user dari table, data akan secara otomatis masuk ke form dan admin dapat mengupdate data tersebut ataupun mendelete user yang dipilih dengan cara menekan tombol update ataupun delete yang ada dibagian bawah form. Form ini juga memiliki validasi sesuai dengan yang diminta pada requirement.

* Admin – Menu items

A screenshot of a computer

Description automatically generated

Cara kerjnya sama dengan page View Users, admin dapat memilih data pada table dan data tersebut otomatis masuk kedalam form. Selanjutnya admin dapat melakukan update ataupun delete terhadap data menu item tersebut. Selain dari update dan delete, form ini dapat digunakan untuk menginsert data juga.

* Customer – Menu Items

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Pada Tampilan menu items ini customer dapat membuat order, customer dapat memilih order item dari table dan akan memasukan quantity yang inginkan, jika quantity = 0 maka tidak akan dimasukan kedalam database, validasi quantity juga berlaku disini dimana quantity tidak boleh kurang dari 0. Untuk menambahkan menu item pada list orderan customer dapat menekan button add item dan untuk meremove item customer dapat milih lagi menuitem yang ingin diremove dari table dan menekan tombol remove item. Ketika customer menekan tombol remove item, item tersebut akan dirubah quantitynya menjadi 0 sehingga tidak akan masuk kedalam database. Dibagian paling bawah terdapat button submit dan clear, setelah customer menambahkan banyak menu item kedalam list orderan customer dapat melakukan order dengan cara menekan button submit order, dan jika semisal customer tidak jadi mesan atau ingin mengulang pemesanan udah dapat menekan button clear order. Note: secara default order yang dibuat akan memiliki status Pending.

* Customer – Active Orders

A screenshot of a computer

Description automatically generated

Pada page ini customer dapat melihat semua order mereka yang masih aktif. Maksud dari masih aktif disini adalah order-order yang memiliki status Pending/Prepared/Served.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Jika salah satu dari Active Orders tadi di tekan, maka customer akan diredirect ke page detail order dimana customer dapat melihat menu item apa saja yang ada pada order tersebut. Selain itu customer juga dapat melakukan update terhadap order mereka (add/remove menu item). Setelah customer melakukan perubahan jika semisal customer merasa tidak jadi ingin merubahnya maka customer dapat menekan tombol revert order dan balik ke order mereka yang awal sebelum perubahan. Customer juga dapat melakukan cancel order dengan cara menekan button cancel order.

* Customer – Order History

A screenshot of a computer

Description automatically generated

Pada page ini customer dapat melihat riwayat order mereka (order yang memiliki statas Paid atau Cancel)

A screenshot of a computer

Description automatically generated

Jika salah satu dari data riwayat order pada table ditekan maka customer akan diredirect ke page yang menunjukan detail dari order tersebut seperti menu item apa saja yang ada pada order tersebut dan juga total harga dari order tesebut.

* Chef – Pending Orders

A screenshot of a computer

Description automatically generated

Pada page ini chef dapat melihat semua pending order yang diorder oleh semua user.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Jika chef menekan salah satu dari pending orders yang ada pada table maka chef akan diredirect ke page yang kurang lebih mirip dengan page User – Active Orders Detail (page Ketika salah satu dari active orders ditekan). Pada page ini chef dapat melakukan perubahan pada order yang bersangkutan seperti add/remove menu item, update order, revert order, cancel order. Ada satu hal special yang hanya bisa dilakukan oleh chef yaitu prepare order. Chef dapat mengubah status dari sebuah order dari yang tadinya bersatatus Pending menjadi Prepared.

* Waiter – Prepared Orders

A screenshot of a computer

Description automatically generated

Pada prepared orders yang dapat diakses oleh waiter ini kurang lebih sama dengan page pending orders yang dapat diakses oleh chef, hanya saja ada sedikit perbedaan, order yang tampil pada table di page ini hanyalah order yang memilki status Prepared, lain dari chef yang dimana order yang tampil hanya yang berstatus pending saja.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Jika suatu prepared order pada table ditekan maka waiter akan diredirect ke page yang kurang lebih mirip dengan page detail dari pending order chef tadi (page yang muncul Ketika suatu pending order ditekan oleh chef). Pada page detail prepared order ini memiliki 1 fungsi yang berbeda yaitu waiter dapat mengubah status order dari yang tadinya Prepared menjadi Served dengan cara menekan button Serve Order. Fungsi selebihnya yang ada dipage ini sama bisa memodifikasi order.

* Cashier – Ongoing Orders

A screenshot of a computer

Description automatically generated

Pada page ini cashier dapat melihat semua order yang sedang berjalan (order yang memiliki status Pending, Prepared, Served). Note: disini saya berasumsi bahwa pembayaran dapat dilakukan kapanpun sebelum atau setelah makan sesuka customernya.

A screenshot of a computer

Description automatically generated

Jika salah satu dari ongoing orders itu ditekan oleh cashier maka cashier akan diredirect ke sebuah page yang menampilkan detail menu-menu item yang dipesan oleh order tersebut dan cashier dapat memproses pembayaran melalui form yang ada pada page ini. Process pembayaran dapat dilakukan dengan cara mengisi form yang ada dengan memenuhi semua validasi yang ada sepesrti payment type harus berupa Cash, Credit, ataupun Debit dan untuk harga yang dibayar itu tidak boleh kurang dari total price order tersebut. *Untuk harga disini saya berasumsi bahwa untuk harga yang melebihi harga pokok(sum dari quantity\*price dari semua menu item) maka harga lebihnya tersebut menjadi biaya lain-lain termasuk tips.* Jika semua validasi sudah terpenuhi maka cashier sudah dapat memproses pembayaran untuk order tersebut dengan cara menekan buttom Process Payment. Dengan ditekannya button process payment maka terbentuklah Receipt dari order tersebut dan order tersebut juga akan berubah statusnya dari yang tadinya Pending/Prepared/Served akan menjadi Paid.

* Cashier – Receipts

A screenshot of a computer

Description automatically generated

Pada page receipt ini cashier dapat melihat semua receipt yang sudah pernah terbentuk.

A screenshot of a computer

Description automatically generated

Jika sebuah receipt dari table ditekan oleh cashier maka cashier akan diredirect ke page yang menampilkan menu-menu item yang dipesan oleh order yang bersangkutan. Selain itu cashier juga dapat melihat Date, Payment Type, dan Total Price juga.

Sekian penjelasan code dan juga tampilan dari saya, semoga sudah cukup jelas, terima kasih.

Note Penting: Bersamaan dengan file dokumentasi ini terlampir juga file project, file sql dan file txt yang berisi query sql manual. Untuk menjalankan aplikasi diwajibkan untuk mengcreate database dengan nama ‘mysticgrills’ terlebih dahulu dan mengimport file sql, atau menjalankan query sql manual dari file txt sesuai petunjuk yang ada pada file txt tersebut, terima kasih.

* Reference
* Materi Kelas
* Group Member
* 2501989784 – David Joan