**Live Order**

**A**

**Project Report**

**Submitted by**

***KEVAL* DHOLAKIYA *& VAITUL BHAYANI***

**FOR FULFILLMENT OF FIFTH SEMESTER IN**

**BACHELOR OF COMPUTER APPLICATION**

****

**Department of Computer Science & I.T.**

SHREE H.N.SHUKLA

GROUP OF COLLEGES

Our Profile



Name: Keval Dholakiya

Stream: BCA 5

Mobile: +91 7096 823 708

Email: kevald47@gmail.com



Name: Vaitul Bhayani

Stream: BCA 5

Mobile: +91 9904021519

Email: personal.vaitul@gmail.com

Preface

Theory of any subject is important but without its practice it useless particularly for the computer student. A computer student cannot become perfect man of technologist without practical understanding of branch.

We can also say, “Experience is the best teacher” so our project is one Kind of Experience, the part of our life. Through the project we have learnt Good, Real, and Practical application.

The project training in the B.C.A of the course gives us the exposure to real world. The aim of the project training, by understanding a project, is to have practical experience of the real world. It also clears the picture of practical field to prompt the student to develop their qualities talents, etc. So that we can expand our view about the implication of the theoretical Knowledge in the practical field, we had the opportunity for the same at various field.

Acknowledgement

The present of this report gives us the feeling of the final frontier toward achieving the BCA Degree; the activity of going through industrials orientation has bridged the gap between the academics and practical real-life work for us.

It has prepared us to apply ourselves to become good IT professional. Naturally, it requires lot of people support to complete this project. We take this opportunity to acknowledge their support to us.

In the acknowledge, Hear first of all we would like to thanks our College Shree H.N Shukla that they gave such Good opportunity for develop a project which would evaluated. We Thanks to our friends to give support for a development of this project.

We would also like to thanks all those people who made this project directly or indirectly possible.

*Thank You*

Keval H. Dholakiya

Vaitul B. Bhayani

Index

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Title** | | **Page No.** | |
|  | | [Introduction of Project](#IntroductionofProject) | | 05 | |
|  | | [Project Profile](#ProjectProfile) | | 08 | |
|  | | [Data Dictionary](#DataDictionary) | | 09 | |
|  | | [Data Flow Diagram](#DataFlowDiagram) | | 11 | |
|  | | [Use Case Diagram](#UsecaseDiagram) | | 13 | |
|  | | [E-R Diagram](#ERDiagram) | | 14 | |
|  | | [System Design](#SystemDesign) | | 15 | |
|  | | [Testing](#Testing) | | 20 | |
|  | | [Conclusion](#Conclusion) | | 21 | |
|  | | [Bibliography](#Bibliography) | | 22 | |

Introduction of Project

We Created Project Named “*Live Order*”, Here We Provide A Complete Solution For Waiter And Chef in Restaurants.

Our Main Goal is to digitalize working of restaurant by smart phones or tablets so Waiters and Chef of restaurant can done things in fewer efforts in less time.

Here is How It Works,

* First of All a Customer Walks In Restaurant.
* Waiter Takes Order On Smart Device.
* As Soon As Waiter Takes Order, It Directly Informed In Kitchen Room So Chef Start Preparing Meal.
* After Meal was Prepared Chef Mark as Cooked on Smart Device.
* Waiter Get List of Cooked Meal and Table Number of Order.
* If Waiter-1 is Free so, he takes cooked meal From Kitchen. As Soon as Waiter Takes, Meal was Removed from Waiting List and Added to Served List.
* After Customer Finished, Waiter Request Fro Get Bill On Device And Waiter Get Full List Of Customer Order. After Being Paid, Customer Have Option To Get Bill On her/his Email. In addition, Bill Get Print of Wireless Printer.
* Here, we Give Discount to Frequent Customer, Who Come in Restaurant 10 or More Time.
* There’s also an admin Panel Where Owner Or Admin Have Full Control On System,
* Admin Can, Add, Edit, Remove Chef And Waiters And Also Add, Edit, Remove Products And Table

Our Website is Live at <http://liveorder.tk> you can visit

Credentials Are

* For Waiter // username = tonystark and password = tonystark
* For Chef // username = batman and password = batman
* For admin // username = superman and password = superman

SDLC

We use incremental model for our project

* Requirement Gathering :

For Requirement Gathering We Took Help of Our Foodie Friend Who used to visit various type of restaurant. We Ask him about problems that he faces when he go to restaurant. Moreover, we come to know that almost all restaurant have service problem they delay in provide food. So We Decide To Solve That Problem By Digitalize Their System.

* Design :

After Analyze Problem We Decide To Design 3 Module Of System. Waiter, Chef and Admin All Three Are main aspect of Restaurant. We decide to use core php for backend and html, css, JavaScript for frontend moreover MySQL for database. We Also Use GitHub for Version Control System and Collaboration Between us.

* Development :

After deciding which technology we will use. We start developing. After 2 Month, We Developed Project. Now we have to test it.

* Testing:

We Host Our Site To Enable our Friends To Give Feedback About Project and Inform If Anyone Finds Bug.

* Deployment :

After Analyze Valuable Feedback of Our Friend and We Fix Bugs/Errors and then we Release our webapp officially. We use free site hosting 000webhost to host our site.

* Maintenance :

After Releasing, We Continue Maintaining Our Site. If Some One Reports Us About Bug Or Anything. We try to solve it.

Project Profile

Project Name : Live Order

Front End : HTML, JavaScript, CSS

Back End : PHP, MySQL

Libraries : JQuery, JQueryUI & Bootstrap

Database : MySQL

Documentation Tool : Microsoft Word 2019

Platform : Web

Web Server : Apache

Web Browser : Any except Windows IE

Source Code / VCS : https://github.com/liveorder

Project Duration : 1 Month

Prepared By : Keval Dholakiya & Vaitul Bhayani

Submitted To : Shree H.N Shukla College

Data Dictionary

* **Table Name**: users
* **Table Description**: Store All Info About users

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Constraints | Description |
| Id | Int(2) | Primary Key | - |
| FirstName | varchar(50) | Nullable | First Name of User |
| LastName | varchar(50) | Nullable | Last Name of User |
| Username | Varchar(50) | Not Null | Username of User |
| Password | Varchar(50) | Not Null | Password of User |
| Role | enum('Waiter', 'Chef', 'Admin') | Not Null | Role of User |

* **Table Name**: Products
* **Table Description**: Store All Info Products/Items of Each Category

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Constraints | Description |
| Id | Int(2) | Primary Key | - |
| catId | int(2) | Not Null | Category Id for this Product |
| Name | varchar(50) | Not Null | Product/Items Name |
| Price | double | Not Null | Price of product |
| IsAvailable | enum('1', '0') | Not Null | Check items available or not |

* **Table Name**: tables
* **Table Description**: Store All Info Tables

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Constraints | Description |
| Id | Int(2) | Primary Key | - |
| Name | varchar(50) | Nullable | Table Name |
| Capacity | int(2) | Nullable | Total Capacity of Chairs |
| IsOccupied | tinyint(1) | Not Null | Describes table is occupied or not |

* **Table Name**: InvoiceItems
* **Table Description**: Store All invoice sub items which is ordered by customer

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Constraints | Description |
| Id | Int(2) | Primary Key | - |
| InvoiceId | int(3) | Not Null | Contains Invoice Id |
| ProductId | Int(3) | Not Null | Describes Product was ordered |
| Quantity | Int(2) | Not Null | Total Quantity of product |
| Rate | Double | Not Null | Rate of Product |
| Amount | Double | Not Null | Amount = Qty \* Rate |

* **Table Name**: invoices
* **Table Description**: Store invoice data with gst

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Constraints | Description |
| Id | Int(2) | Primary Key | - |
| time | timestamp | Not Null | Add current timestamp automatically |
| tableId | Int2) | Not Null | Generate invoice for this table |
| GrossAmount | Double | Not Null | It is Gross Amount |
| GSTP | Double | Not Null | GST % |
| GSTRs | Double | Not Null | GST Rs. |
| TotalAmount | Double | Not Null | Total Amount |
| Waiter | Varchar(255) | Not Null | Waiter Name who generating invoice |

* **Table Name**: kitchen
* **Table Description**: Store info about kitchen

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Constraints | Description |
| Id | Int(2) | Primary Key | - |
| time | timestamp | Not Null | Add current timestamp automatically |
| tableId | Int(3) | Not Null | For which table were order come from |
| ProductId | Int(3) | Not Null | It is Product will be cook |
| Quantity | Int(2) | Not Null | Quantity of ordered item |
| Pending | Int(2) | Not Null | For how many items not ready |
| isReady | Tinyint(2) | Not Null | For items which is Cooked |

* **Table Name**: categories
* **Table Description**: Store All Info About Products Category

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Constraints | Description |
| Id | Int(2) | Primary Key | - |
| Name | varchar(255) | Nullable | Describes name of Product Category |

Data Flow Diagram

* 0 Leval DFD

Customer

Waiter

Admin

Chef

* 1 Level DFD

Serve Meal

Get Order

Chef

Give Invoice

Generate Invoice

Send Order Data

Meal Served

Waiting

Take Meal

Take Order

Analyse

Give Order

Serve Meal

DataBase

Admin

Waiter

Customer

Use case Diagram

Waiter

Chef

ER Diagram

**Users**

**Products**

**Categories**

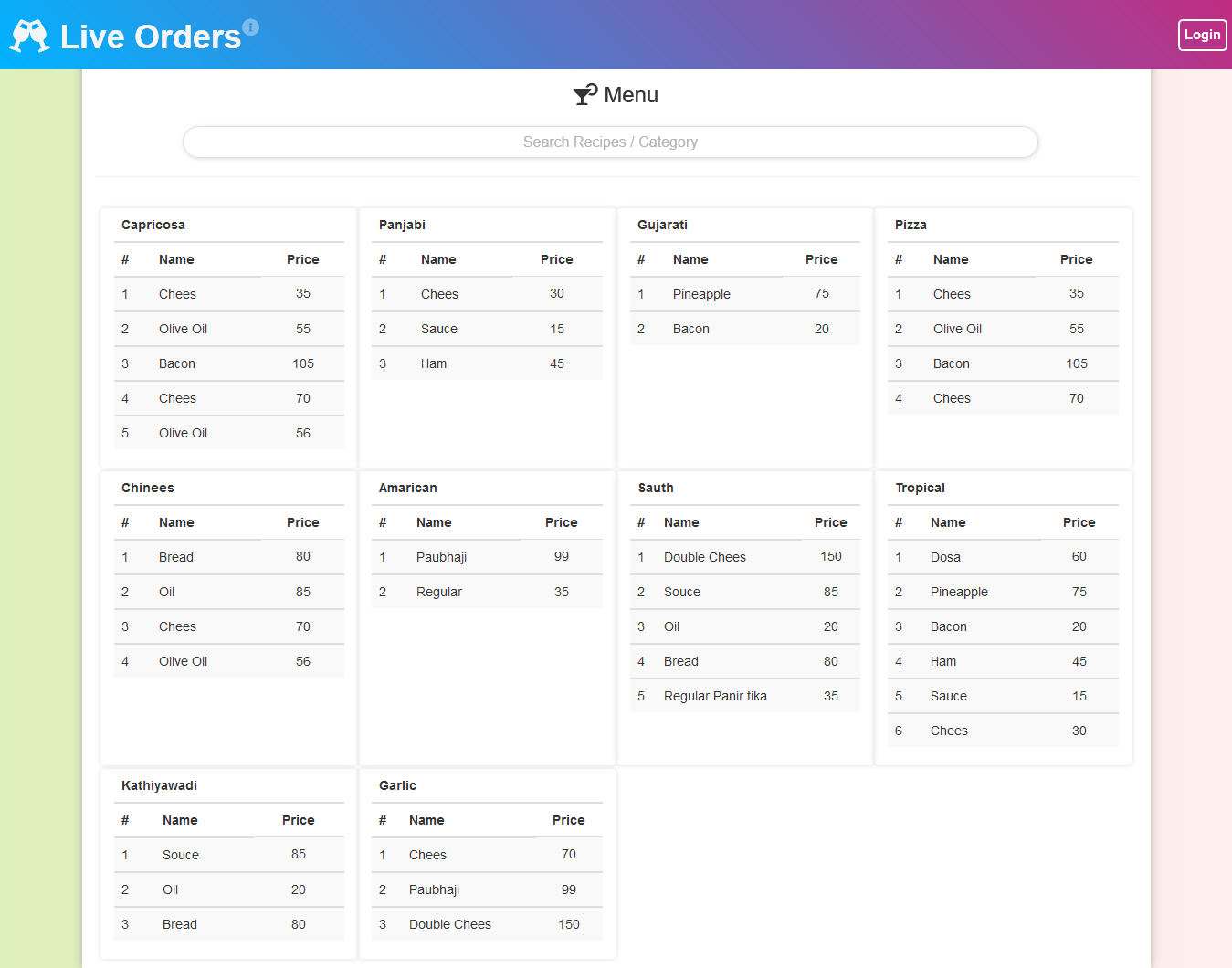
**Item**

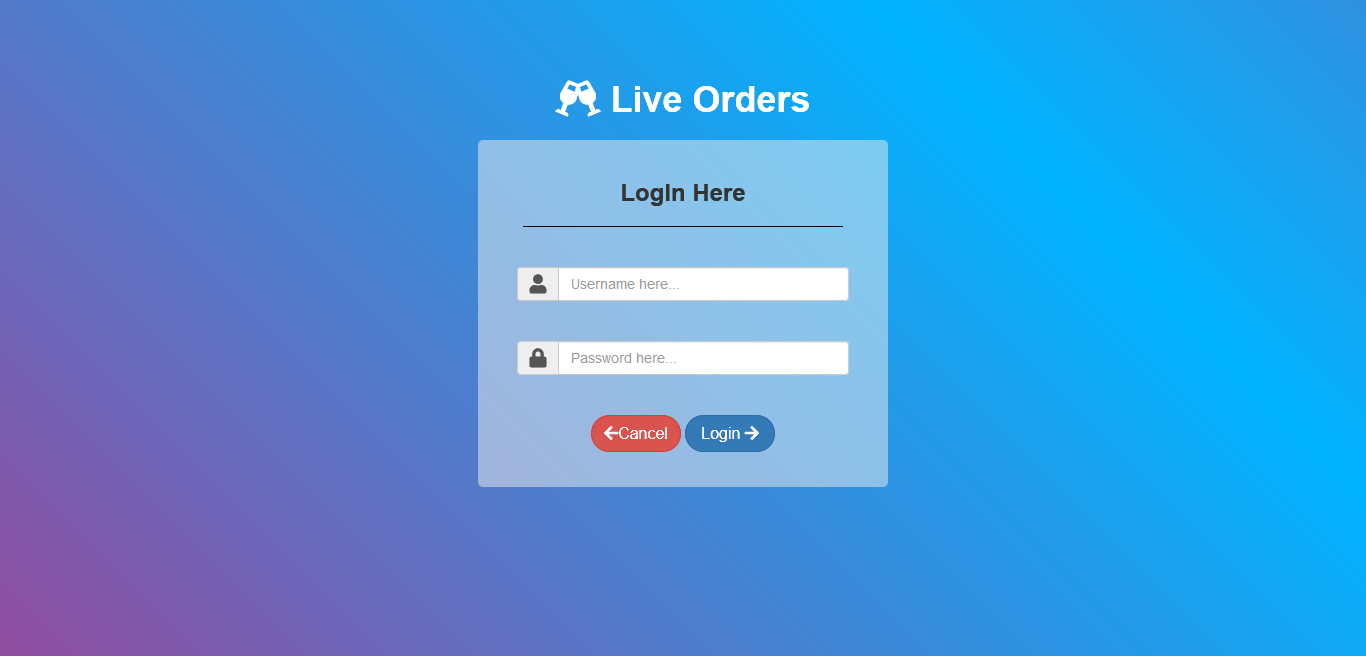
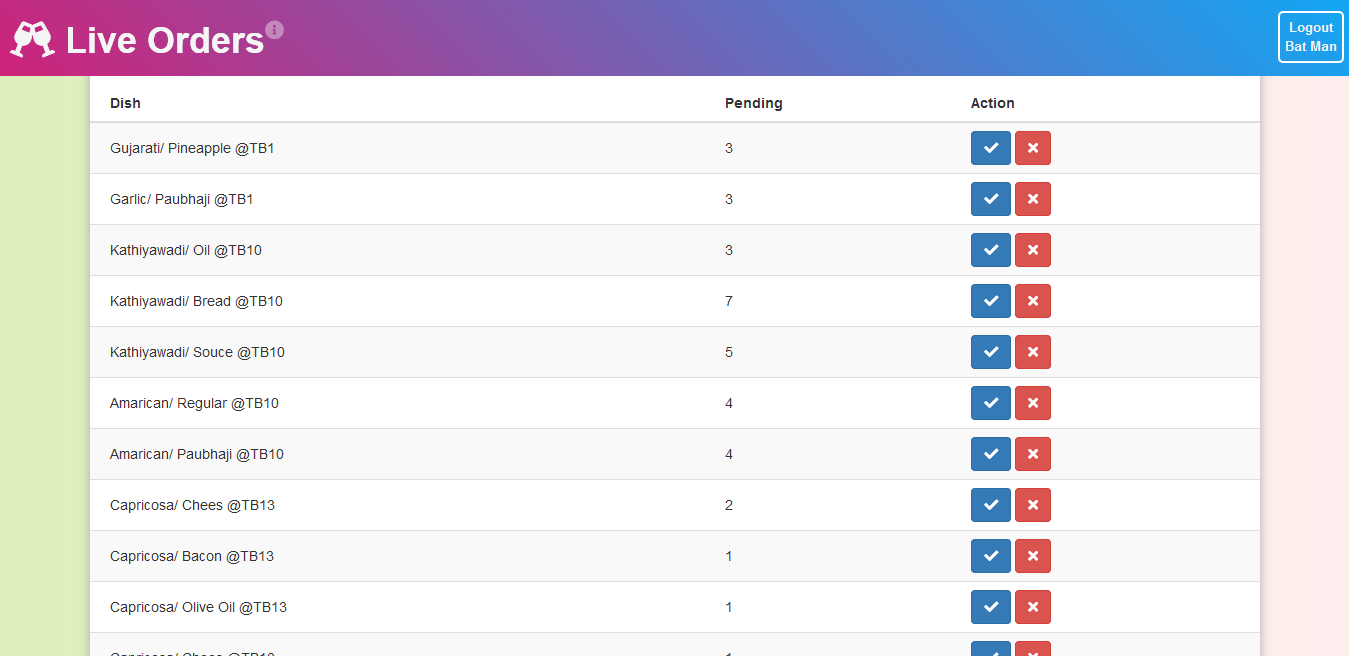
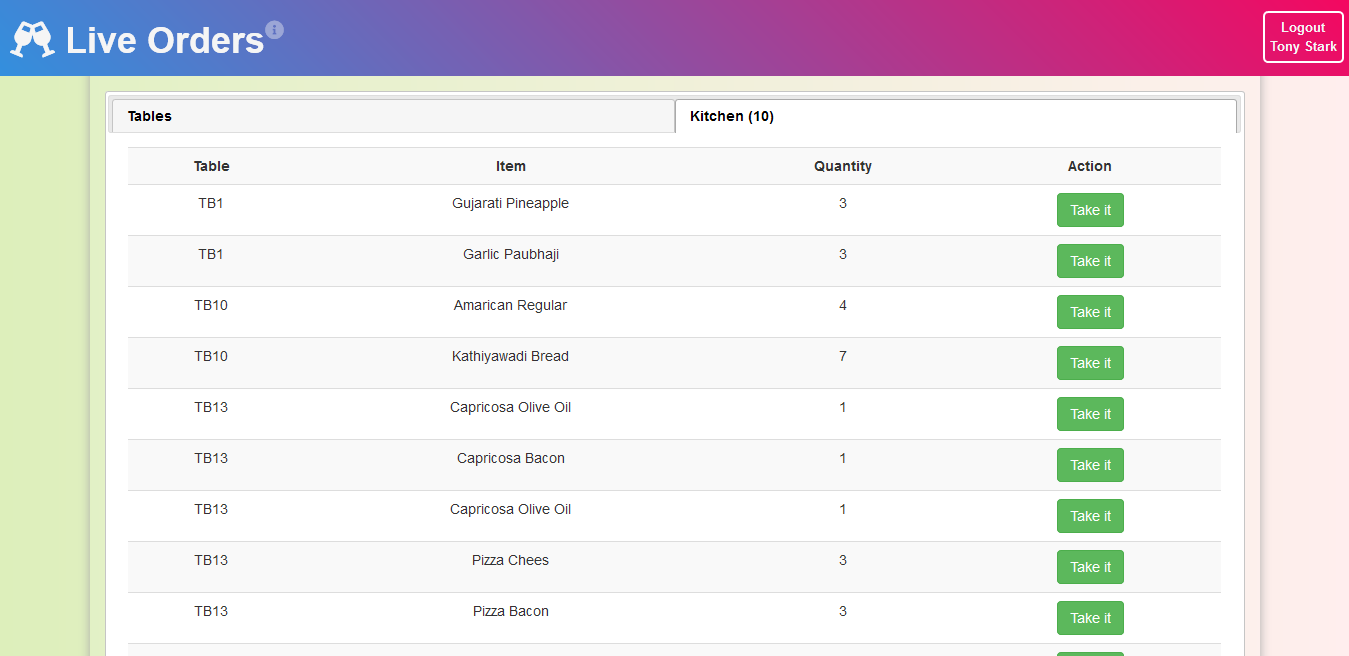
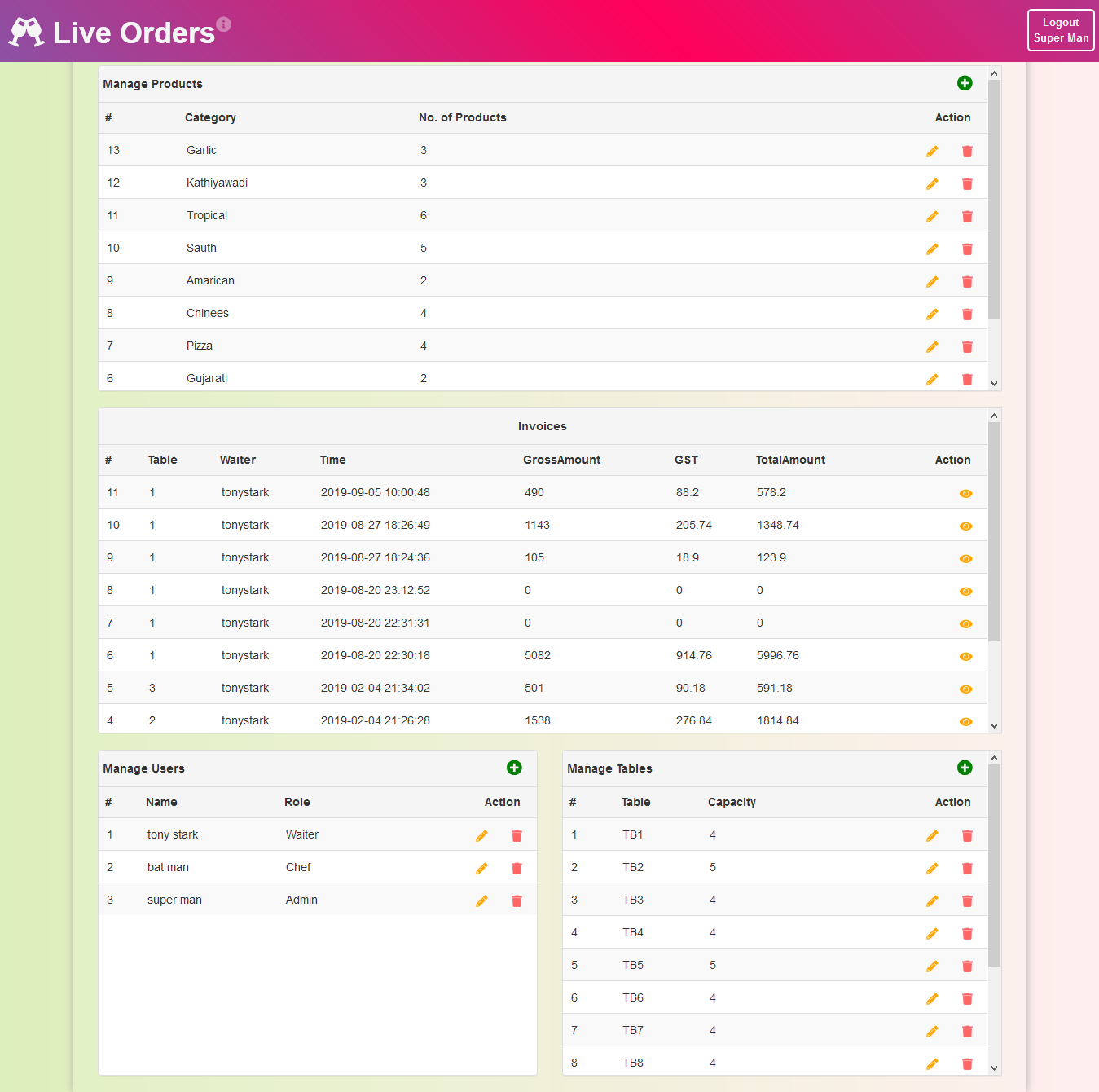
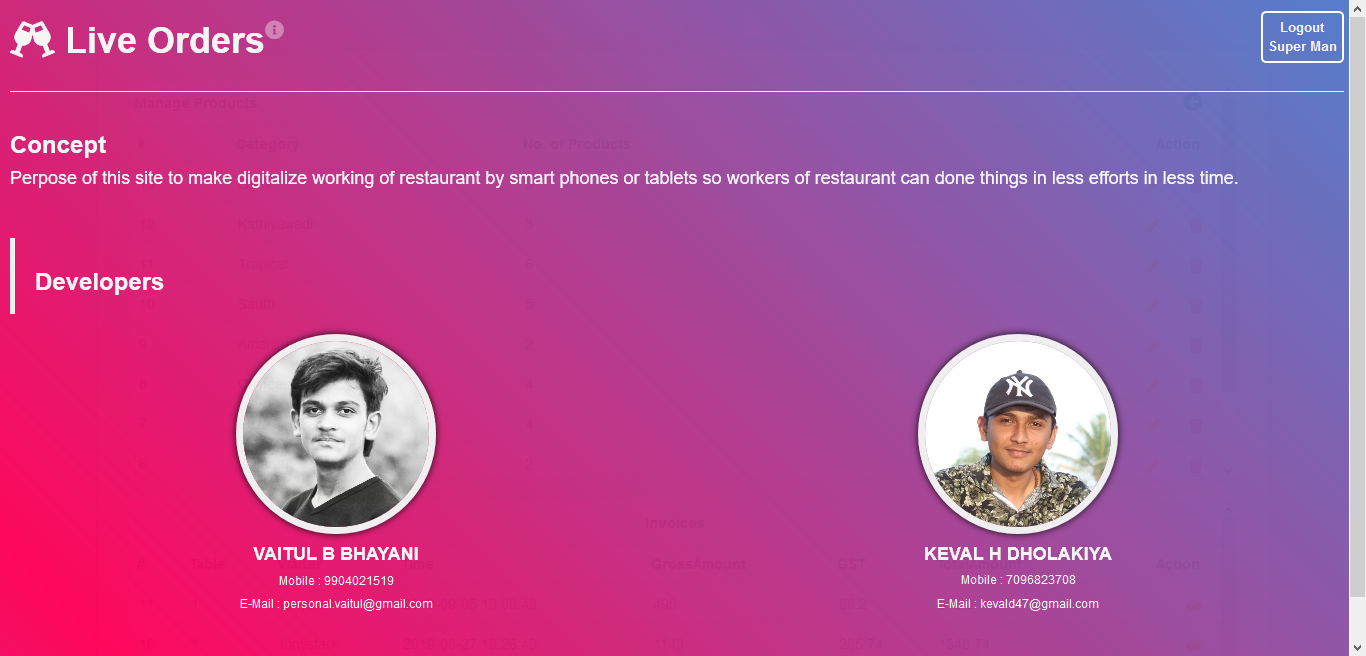
**Invoice**

**Kitchen**

System Design

* View : Home Screen
* Desc : Display Menu which for Customer, he/she can see available items



* View : Login Screen
* View : Chef Screen
* Desc: Display Orders.
* Page : Waiter Screen (tables)
* View : Waiter Screen (Kitchen)
* View : Admin Screen
* Desc : Display Admin Module
* View: About us

Testing

**Manual**

We use **Black Box** testing technique to test this site

We tested several times and we sorted out some issue which is we identified.

In addition, we launched Beta version for our College colleague to test and give the feedback or Suggestion to us.

We Also Perform **White Box** Testing by Our Selves and Our Colleagues Who have Experience of Web Developing.

Conclusion

The project report entitled "Live Order" has come to its conclusion

The new system has been developed with so much care that it is free of errors and at the same time, efficient and less time consuming

System is robust. In addition, provision is provided for future developments in the system

It was a wonderful learning experience for us while working on this project. This project took us through the various phases of project development and gave us a real insight into the world of software engineering. The joy of working and the thrill involved while tackling the various problems and challenges gave me a feel of developers industry.

It was due to this project we came to know how professional software is designed.

In this project we learned a lot we thankful to our college and Saurashtra University.

Bibliography

Here are the following names of website referred during the different system development life cycle. We referred this book and website at the time during difficulties and doubts. Really those website are surely our best friends at the time of problems, they help a lot.

* Websites References:

• [www.stackoverflow.com](http://www.stackoverflow.com)

• [www.w3schools.com](http://www.w3schools.com)