Contents

[1. INTRODUCTION 3](#_Toc479849778)

[1.1 Project Summary 3](#_Toc479849779)

[1.2 Purpose 3](#_Toc479849780)

[1.3 Scope 3](#_Toc479849781)

[1.4 Literature Review 3](#_Toc479849782)

[2. PROJECT MANAGEMENT 4](#_Toc479849783)

[2.1 Project Planning and Scheduling 4](#_Toc479849784)

[2.1.1 Project Development Approach 4](#_Toc479849785)

[2.1.2 Project Plan 5](#_Toc479849786)

[2.1.3 Schedule Representation 5](#_Toc479849787)

[2.2 Risk Management 6](#_Toc479849788)

[2.3 Estimation 7](#_Toc479849789)

[2.3.1 Effort Estimation 7](#_Toc479849790)

[Name 9](#_Toc479849791)

[2.3.2 Cost Estimation 9](#_Toc479849792)

[3. SYSTEM REQUIREMENTS STUDY 12](#_Toc479849793)

[3.1 User characteristics 12](#_Toc479849794)

[3.2 Hardware and Software Requirements 12](#_Toc479849795)

[3.3 Constraints 12](#_Toc479849796)

[4. SYSTEM ANALYSIS 13](#_Toc479849797)

[4.1 Comparison with existing systems 13](#_Toc479849798)

[4.2 Functional requirements 15](#_Toc479849799)

[4.3 Non-functional requirements 19](#_Toc479849800)

[4.3.1 Reliability 19](#_Toc479849801)

[4.3.2 Availability 19](#_Toc479849802)

[4.3.3 Security 20](#_Toc479849803)

[4.3.4 Maintainability 20](#_Toc479849804)

[4.4 Functions of system 20](#_Toc479849805)

[4.4.1 Use case diagram 20](#_Toc479849806)

[4.5 Data modeling 23](#_Toc479849807)

[4.5.1 ER Diagram 23](#_Toc479849808)

[4.5.2 Data flow diagram 24](#_Toc479849809)

[4.6 Functional and behavioral modeling 28](#_Toc479849810)

[4.6.1 Control flow diagram 28](#_Toc479849811)

[5. SYSTEM DESIGN 29](#_Toc479849812)

[5.1 Architecture diagram 29](#_Toc479849813)

[5.2 Component Diagram 30](#_Toc479849814)

[6. TESTING 31](#_Toc479849815)

[7. FUTURE SCOPE 40](#_Toc479849816)

[• Packed corporate meals 40](#_Toc479849817)

[• Catering for events 40](#_Toc479849818)

[• Website for greater reach 40](#_Toc479849819)

[• Third party delivery service 40](#_Toc479849820)

[• Automated authentication of delivered tiffins 40](#_Toc479849821)

[8. CONCLUSION 40](#_Toc479849822)

[APPENDIX 41](#_Toc479849823)

[Prototype 41](#_Toc479849824)

# 1. INTRODUCTION

## 1.1 Project Summary

* Tiffiny is a tiffin service portal which lists the tiffin providers in a user’s area and makes it easier for the customers to find the best tiffin services in their area.
* It also enables easy ordering through the app where it only takes a click.
* The users can customize their meal and change the portions or the items they order, freeing the customer from the mandate of having to buy four things even when he wants to eat only two.
* They can pay through COD or Payment gateways such as Paypal or Freecharge.
* The payment information also stores the number of people and the people the tiffin is being shared with, making it easier to split money and unburdening the user(especially the hostel students) from remembering who shared which tiffin and who owes how much.
* As for the vendor, the app given him custom analytic services based on the overall market data and his own data. This is to bring

## 1.2 Purpose

* The purpose of the app is to make tiffins more accessible and give a broader exposure to vendors.
* It intends to make food a simple matter in hostels and make it easier for people to find good food.
* For the vendors it aims to provide them with a larger toolset to make their services better.
* It also intends to provide the vendors with analytical insights in a graphical manner making analytics accessible to people without .

## 1.3 Scope

As more and more people move away from their homes in pursuit of better opportunities, food becomes a top priority. Gastronomic service aggregators such as foodPanda and Zomato offer a platter of options but all too costly and unsuitable for everyday meals.

In such a scenario, the Indian custom of tiffin is a viable option worth reviving.

## 1.4 Literature Review

* The existing services are limited to listing the tiffin providers and ordering.
* They allow a tiffin vendor monopoly where the customer has no flexibility or choice in the items.
* This system intends to aggregate the listing model along with customization to make the tiffin service model customer centric.
  + Along with that, the problems of sharing tiffin such as who did a person share it with, who owes how much are solved through this system.

# 2. PROJECT MANAGEMENT

## 2.1 Project Planning and Scheduling

### 2.1.1 Project Development Approach

The basic idea is to build software product iteratively by adding several new features during each increment of the product. Requirements are well understood and specified. Each iteration includes new features from requirements. Therefore, progress of the project can be tracked with the help of increments and each increment can be developed within short development cycle.

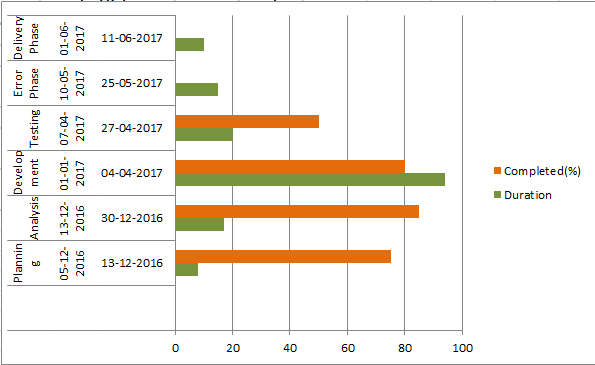
Since requirements are well understood it is easy to decide what new features will go in new upcoming increment of the product. This approach allows early release of product with limited set of features which can be improved in upcoming increments.



### 2.1.2 Project Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Starting Date | End Date | Duration  (in days) | Completed(%) |
| Planning | 01-10-2016 | 13-12-2016 | 8 | 75 |
| Analysis | 13-12-2016 | 30-12-2016 | 17 | 85 |
| Development | 01-01-2017 | 4-04-2017 | 94 | 80 |
| Testing | 07-04-2017 | 27-04-2017 | 20 | 50 |
| Error Phase | 10-05-2017 | 25-05-2017 | 15 | 0 |
| Delivery Phase | 01-06-2017 | 11-06-2017 | 10 | 0 |

### 2.1.3 Schedule Representation



## 2.2 Risk Management

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Risk | Prob  ab  -ility | Impa  ct | Effecton  Project | Risk  Reduction  Action | If ithappens  Triggers  &Action |
| 1 | Significant  changes in user requirements | Low | High | Time-  quality-cost | Ensurethat theuser requirements arefully investigated andagreed before specification | **Triggers**  Requestfor changes to agreed  specification  **Actions**  Discuss impactof changeon  schedules or  design, and agreeif changeto specification willproceed.  Implement project change, ifagreed. |
| 2 | Lackof Academic and Departmentalusage | High | High | Failure to achieve business benefits.  Ineffective work practices. More fragmented | Ensure User Requirement s are  properlyassessed. Executive leadership  andongoing  involvement. Communications and planning | **Triggers**  Staff Survey, Benefits realizationmonitoring  **Actions**  Review deliverables |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | processes.  Poor Communication. | focus.  Appoint CommsManager |  |
| 3 | Loss of power or wifi connection at key roll-out, demo or training events | Low | Low | Users will not take the product seriously if it cannot be demonstrated to work. | Check for scheduled outages. | **Actions**  If doing demos/training, have screenshots available locally in case of loss of power. |

## 2.3 Estimation

### 2.3.1 Effort Estimation

**Estimation on Size (FP → KSLOC)**

**Table 2 Function Point Estimation for E-Commerce**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | External User Types | Complexity | FP |
| Login/Signup | External User Input | Low | 3 |
| Create a new tiffin | External User Input | Average | 4 |
| Update/Delete a tiffin | External User Input | Average | 4 |
| Tiffin Description | External User Output | Low | 4 |
| Hamburger menu | External User output | Low | 4 |
| User Profile | External User output | Low | 4 |
| Vendor Profile | External User output | Low | 4 |
| Wallet | External User output | Average | 5 |
| History/logs | External User output | Low | 4 |
| Post payment | External User output | High | 7 |
| Profile update | External User Input | Average | 4 |
| Favourites | External User output | Low | 4 |
| List | External User output | Low | 4 |
| Combo | External User output | Low | 4 |
| View today’s order | External User output | Average | 5 |
| Customize | External User input | High | 6 |
| Payment page | External User Interface | High | 10 |
| Search | External User inquiry | Average | 4 |
| Filter | External User inquiry | Average | 4 |
| Home | External User output | Average | 5 |

Total function points: **93**

Converting to KSLOC in Java (From published figures: 1 FP = 53 SLOC in Java)

Estimated Size: **4.929 KLOC**

**Estimation of Scale Factor:**

**Table 3 Ratings on Scale Factor**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name | Very low  (0.05) | Low  (0.04) | Nominal  (0.03) | High  (0.02) | Very High  (0.01) | Extra High  (0.00) | Assessment | Value |
| Precedentedness | Thoroughly unprecedented | Largely unprecedented | Somewhat unprecedented | Generally familiar | Largely familiar | Thoroughly familiar | Nominal | 0.03 |
| Flexibility | Rigorous | Occasional relaxation | Some relaxation | General conformity | Some conformity | General goals | Very high | 0.01 |
| Significant risks eliminated | Little (20%) | Some (40%) | Often (60%) | Generally (75%) | Mostly (90%) | Full (100%) | Nominal | 0.03 |
| Team interaction process | Very  difficult | Some difficult | Basically cooperative | Largely cooperative | Highly cooperative | Seamless interactions | High | 0.02 |
| Process maturity | Level 1 | Level 2 | Level 2+ | Level 3 | Level 4 | Level 5 | Low | 0.04 |
|  | | | | | | | Add | 1.01 |
| Total | 1.14 |

**Estimation of Effort Adjustment Factor:**

**Table 4 Ratings on Effort Adjustment Factors**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identifier** | Name | **Ranges**  **(VL – EH)** | **Assessment**  **VL/L/N/H/VH/EH** | **Values** |
| RCPX | product Reliability and ComPleXity | 0.5 – 1.5 | Extremely High | 1.5 |
| RUSE | required reusability | 0.5 – 1.5 | nominal | 1.0 |
| PDIF | Platform DIFficulty | 0.5 – 1.5 | high | 1.1 |
| PERS | PERSonnel capability | 1.5 – 0.5 | high | 0.75 |
| PREX | PeRsonnelEXperience | 1.5 – 0.5 | very high | 0.65 |
| FCIL | FaCILities available | 1.5 – 0.5 | nomial | 1.0 |
| SCED | SChEDule pressure | 1.5 – 0.5 | low | 1.2 |
|  | | | Product | 1.2326 |

The effort estimation of tiffin portal is:

Effort = 2.45 × (2.89)1.14 × 0.4826 = 9.96 person-months

**Post-Architecture Stage:**

* The estimation process is the same as those in the early design stage. The only difference is that it has more EAF to estimate.

### 2.3.2 Cost Estimation

* Wehave createdachat android application.
* The application consistsof seventeenfiles written inJAVA and three files written in XML.
* All pages werefullydesigned to havedifferent contentand perform differenttasks.
* Then, theywerecoded and connected togetheraccordingto theirdesign and tasks.
* **Sub-model Used**: BasicCOCOMOI
* **Model Used**: Organic.
* **Formula Used:**

|  |  |
| --- | --- |
| Effort(Man/Month)=32 x (KLOC)1.05 | ------------------------(1) |
| Time=2.5 x (Effort).38 | ------------------------(2) |

* **EstimatingEffort:**
  + Effort=32 x(5)1.05
  + Effort = 5.42 MM
* **EstimatingTime:**
  + Time =2.5 x(5.42)0.38
  + Time =4.75 Months.
* **Sub-model Used**:Intermediate COCOMOI
  + **Model Used**: Organic
  + **FormulaUsed:**
    1. Effort(Man/Month)=EFA x32x(KLOC)1.05
    2. Time=2.5 x(Effort)0.38
* **CostDrivers:**
  + Product Attribute
    1. RELY – Required SoftwareReliability
    2. CPLX-ProductComplexity.
  + Device Attributes
    1. TIME-ExecutionTime
    2. STOR-MainStorage
    3. VIRT-Virtual MachineVolatility
    4. TURN- TurnAroundTime
  + Personal Attributes
    1. ACAP-AnalystCapability
    2. AEXP-ApplicationExperience
    3. PCAP-ProgrammerCapability
    4. VEXP-Virtual MachineExperience
    5. LEXP-Programming LanguageExperience
  + Project Attributes
    1. MODP-Useof Modern ProgrammingPractices
    2. TOOL-Useof SoftwareTool.
    3. SCED-Required DevelopmentSchedule.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Low | Normal | High | VeryHigh |
| RELY |  |  | 1.15 |  |
| CPLX |  |  |  | 1.30 |
| TIME | 0.85 |  |  |  |
| STOR |  |  | 1.21 |  |
| VIRT |  |  | 1.30 |  |
| TURN |  |  | 1.15 |  |
| ACAP |  |  | 0.86 |  |
| AEXP | 0.80 |  |  |  |
| DCAP |  | 1.0 |  |  |
| VEXP |  |  | 0.90 |  |
| LEXP |  |  | 0.95 |  |
| MODP |  | 1.0 |  |  |
| TOOL |  |  |  | 0.83 |
| SCED | 0.85 |  |  |  |

TABLEI:ESTIMATING COST DRIVERSVALUES

* CalculatingEffort Adjustment Factor(EAF):
* Here allassessment values aremultiplied togetherto determinethe EAF:

EAF = 1.15×1.30×0.85×1.21×1.30×1.15×0.86×0.80×1.0×0.90×0.95×1.0×0.83×0.85(5)

EAF = 1.1

The equation further substitutes as follows:

* Effort (Man Month)=1.1×32 x(5)1.05
* Effort = 5.962 MM
* Time = 2.5 x(4.75)0.38.
* Time =4.52 Month

# 3. SYSTEM REQUIREMENTS STUDY

## 3.1 User characteristics

* The typical user of this application is anyone who has an android smartphone with internet capability. It is assumed that user knows basic smartphone usage and knows how to place orders online.

## 3.2 Hardware and Software Requirements

|  |  |
| --- | --- |
| Hardware | Android smartphone with android 5.0 or above.  1GB of RAM.  1 GHz processor.  Internet connection |
| Software | Android studio  JDK 8 or above  JRE 1.6  Android API 21 or above |

## 3.3 Constraints

While the application consists of no hardpressedconcerns,but in case of transaction failure or partial transactions where the customer pays but the status isn’t reflected in the system database,it is mandatory that mechanisms be employed to ensure that the situation is reflected correctly and the customer is refunded without any human intervention.

# 4. SYSTEM ANALYSIS

## 4.1 Comparison with existing systems

To reverse engineer for our topic that is online Tiffin service, we searched for other competitors in this field and we stumbled upon the following existing services. One way or other, they all were having some issues. We noted them and learned from that.

We observed the services provided and platform of the following companies and here are the points we have observed.

**Bhukkadpanti**

1. Website/app : Website

2 .Services provided : Tiffin service, corporate meals

3. Pre-decided menu : No

4. UI ratings/unique features of UI : ease in planning tiffin

5. USP : Ease of cancelation, carry forwarded meal

6. People's review: Customer friendly

* **Customization is possible but food-items are not specified by names.**
* **As we make changes , It displays the total charge accordingly which is a good feature.**
* **Price of each food-item is displayed which gives clarity to customer.**
* **Customization is possible but food-items are not specified by names.**
* **As we make changes , It displays the total charge accordingly which is a good feature.**
* **Price of each food-item is displayed which gives clarity to customer.**

**Spice Box**

1. Website/app : both

2. Services provided : Can choose from different cuisines

3. Pre-decided menu : No

4. UI ratings/unique features of UI – Can track your order

5. USP : -

6. People's review : Special deals for monthly subscription

**Foodella**

1. Website/app : App

2. Services provided : Can choose from different sabzis and dal

3. Pre-decided menu : Yes

4. UI ratings/unique features of UI : -

5. USP : -

6. People's review : Customer service not friendly

**Klick N Eat**

1. Website/app : App

2. Services provided : Different tiffin services to pick from

3. Pre-decided menu : no

4. UI ratings/unique features of UI : UI is not appealing

5. USP : -

6. People's review : -

**What's New?**

* **Provide logs and history**
* **Detailed menu**
* **All the tiffin service providers of the area to choose from**
* **COD**
* **Combos and deals**
* **Favourites**
* **Guaranteed 30 minute delivery**
* **Vendor ratings by users**
* **Sharable wallet**
* **Delivery flow indicator**
* **Customization**
* **Safe, secure and easy payment methods**
* **No need to download separate apps**
* **Sharable wallet and inbuilt calculator**
* **Guaranteed 30 minute delivery**
* **Hassle free experience**
* **Automated tiffin suggestions based on locality**
* **Discounts and offers**
* **Favorites to bookmark**
* **Greater market size for vendors**

## 4.2 Functional requirements

Vendor:

1. Create

After logging in Vendors see the list of their own tiffinson their homepage.

Options such as create a tiffin, edit an existing tiffin will be visible along with the tiffin list. (More on edit in the following section 3)

They can click on the create button to create/add a new tiffin.

On clicking the create button, a page where all the details of the tiffin are to be entered.

Details are:

1. Name of the tiffin
2. Area
3. Cuisine
4. List of Food items
5. Quantity of each food item
6. Click next
7. If each item is separately available, then its price.
8. Click next
9. Tiffin price
10. If half tiffin is available: enter the items for half tiffin
11. Enter the price for half tiffin
12. Lunch or Dinner or Both: date
13. Closing time for lunch / dinner / both
14. Click on save

Saving would commit all the changes to the database and make the tiffin publicly available to the users.

1. Update
2. A logged in Vendor can see his list of tiffins. Against each tiffin he can see an edit button (preferably a pencil symbol) on clicking which he is taken to a page where the entire tiffin is described. The following details can be updated:
3. Area
4. Food items
5. Quantity
6. Which food items are separately available
7. Cost of each food item if separately available
8. Tiffin price
9. Half/full/both
10. Items for half
11. Lunch/dinner/both
12. Closing time
13. Click on save

Saving would commit all the changes to the database and make the tiffin publicly available to the users.

1. Delete

A logged in Vendor can see his list of tiffins. Against each tiffin he can see an edit button on clicking which he is taken to a page where the entire tiffin is described.

On the description page, there would be two buttons save and delete.

On clicking the delete button, the tiffin would be removed from the database.

However before committing the changes to the database, the vendor would be asked to confirm the deletion via a pop-up message.

1. History

In the hamburger menu, a logged in Vendor will find the following options:

1. Payment
2. History
3. View
4. Tiffin Analysis
5. Vendor Profile

History will contain:

1. A list of all tiffins uploaded by the vendor

The tiffins can be filtered using:

1. It shows the number of orders for each tiffin

2. It shows the ratings for each tiffin

3. It also shows the number of users who have this tiffin as a favourite.

5. Payments

Logged in vendor can view his received payments which will be sorted by date of receiving the payment. The details of the payment shall consist of:

* Number of tiffins sold of each type
* Total Cost of each tiffin.

The Current payroll shall show:

* Tiffins ordered until now
* amount

1. View

View shows the orders of the current day. The default view will consist the tiffins ordered according to the closing time for taking orders

The tiffins can be filtered according to:

1. Area
2. Price
3. Number of orders (ascending/descending

**Users:**

The user on logging in will be asked to enter his geographical position if it isn’t available automatically. Following that he will be shown of most popular tiffins in his geographical area.

There will be a slider displaying offers and combos, on clicking which the user will be redirected to the specific offer /combo page. There will be four icons available a hamburger menu, a filter option and a button that redirects to the combo page and a search button .

The tiffins can be filtered by the following attributes:-

1. Ratings
2. Area
3. Price (default descending, toggle between descending and ascending)
4. Cuisine
5. Half/Full (toggle)

The combo button will take the customer to a page that will enlist all available combos.

The search button will enable the user to search tiffins by vendors and tiffin name.

The hamburger menu will show options such as:-

1. User profile
2. Favourites
3. History/Logs
4. Wallet
5. Other Settting

The following options are possible on behalf of the user:-

1. Search
2. Customize
3. Place Order
4. Payment
5. History

**1. Search**

-Logged in customer can search the tiffins via tiffin name or vendor name.

The results can be sorted according to:

1. Lunch/dinner
2. Combos
3. Ratings
4. Price
5. Locality
6. Cuisine

**2) Customize**

The tiffin can be customized according to

1. Proportions
2. Add/remove individual items if the vendor allows.
3. Pick multiple tiffins

**3) Place Order**

Select address from the profile or add new address

Mention among how many people the tiffin is being shared

If the vendor allows, the order can be a monthly subscription order in which the duration of the subscription along with the start and end date will be mentioned.

On clicking place order, the customer is redirected to payment gateway.

**4) Payment**

The customer can pay via wallet, COD, Coupons or perform online transaction using a payment Gateway.

On choosing a payment gateway, a transaction id will be provided which will be the proof of the transaction.

The transaction status along with the id will be stored in the database and will be visible when the users views his payment history.

On completion of the payment, other than COD, the transaction status along with the order confirmation will be provided.

On failure of a transaction, the order will not be placed.

**5) History/ Log**

Customer can see his past orders and the transaction details by clicking history/log.

Past orders will contain the date of the order, the content of the order, the amount and the people shared with.

Transaction details contain transaction id, order id and status of the transactions

### 4.3 Non-functional requirements

### Reliability

The system, on failure: be it connection or server, will reflect the most recent committed events and return to the last visited checkpoint if the failure is in the middle of an order.

### 4.3.2 Availability

The system is intended to run 24hrs with users being able to acess it at any point. In case of a failure, the temporary data should be backed up in the cache and the system should pick up from the last visited checkpoint in case of failure during placing an order.

### 4.3.3 Security

Every login must be stored in the logs. The content is only available to users who are logged in.

Vendors cannot see the overall inventory of the system that the customer could. The view of the vendor is limited to his own uploads and their performance and received orders.

### 4.3.4 Maintainability

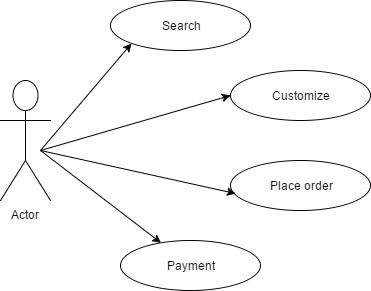
Avoid redundant interfaces, if something can be done in one step, don’t make the user take two.Reusability of code and modularity are emphasized upon.

# 4.4 Functions of system

4.4.1 Use case diagram

tiffiny

Vusecase



User case

## 4.5 Data modeling

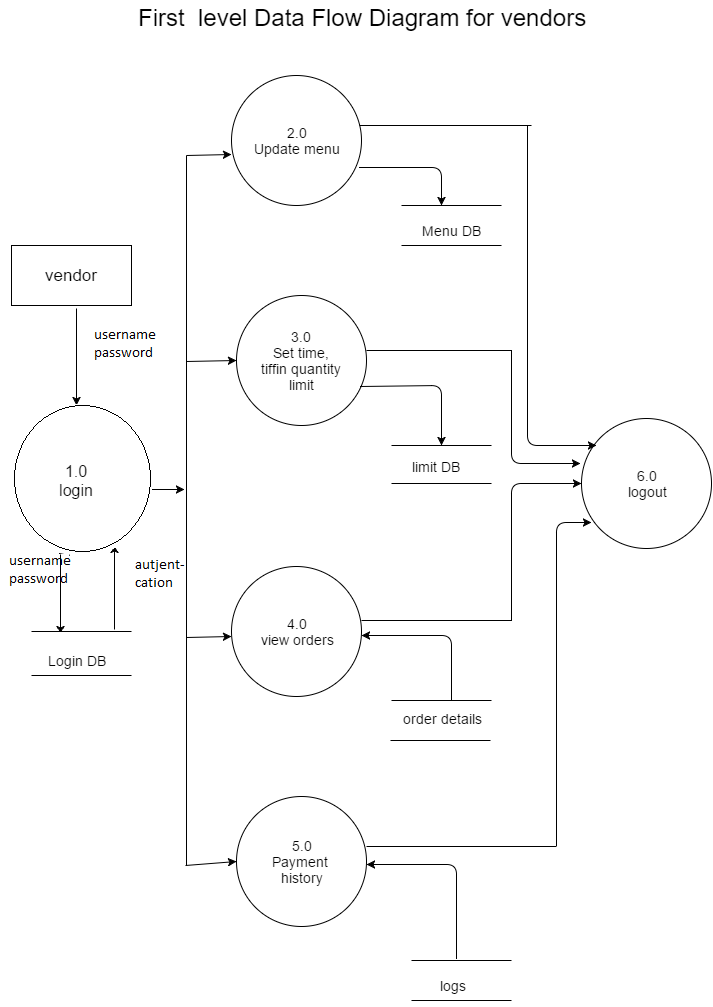
### 4.5.1 ER Diagram

Tiffiny ER

### 4.5.2 Data flow diagram

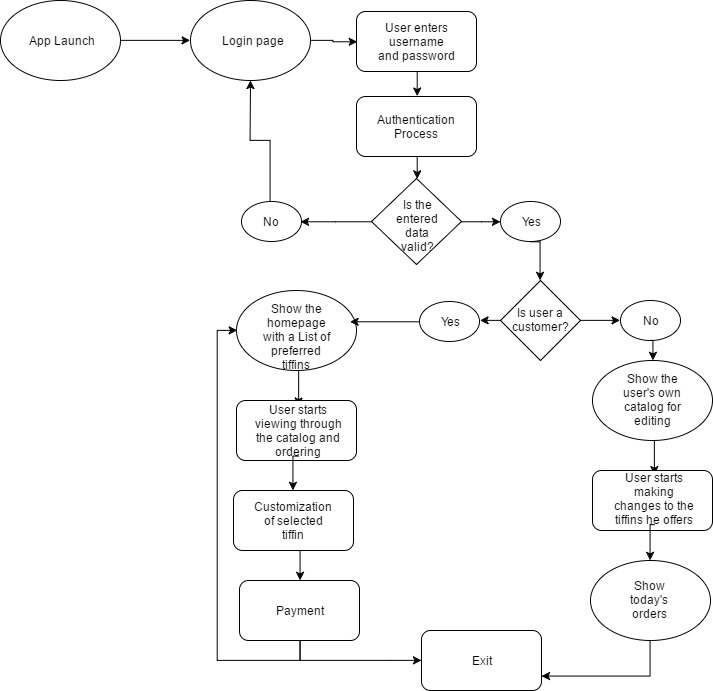
Level 0

**D:\Academics\sem 5\Design Engineering\diagram\context level dfd.png**

Level 1**D:\Academics\sem 5\Design Engineering\diagram\1 dfd Customer.pngD:\Academics\sem 5\Design Engineering\diagram\1 DFD.png**

4.6 Functional and behavioral modeling

4.6.1 Control flow diagram

****

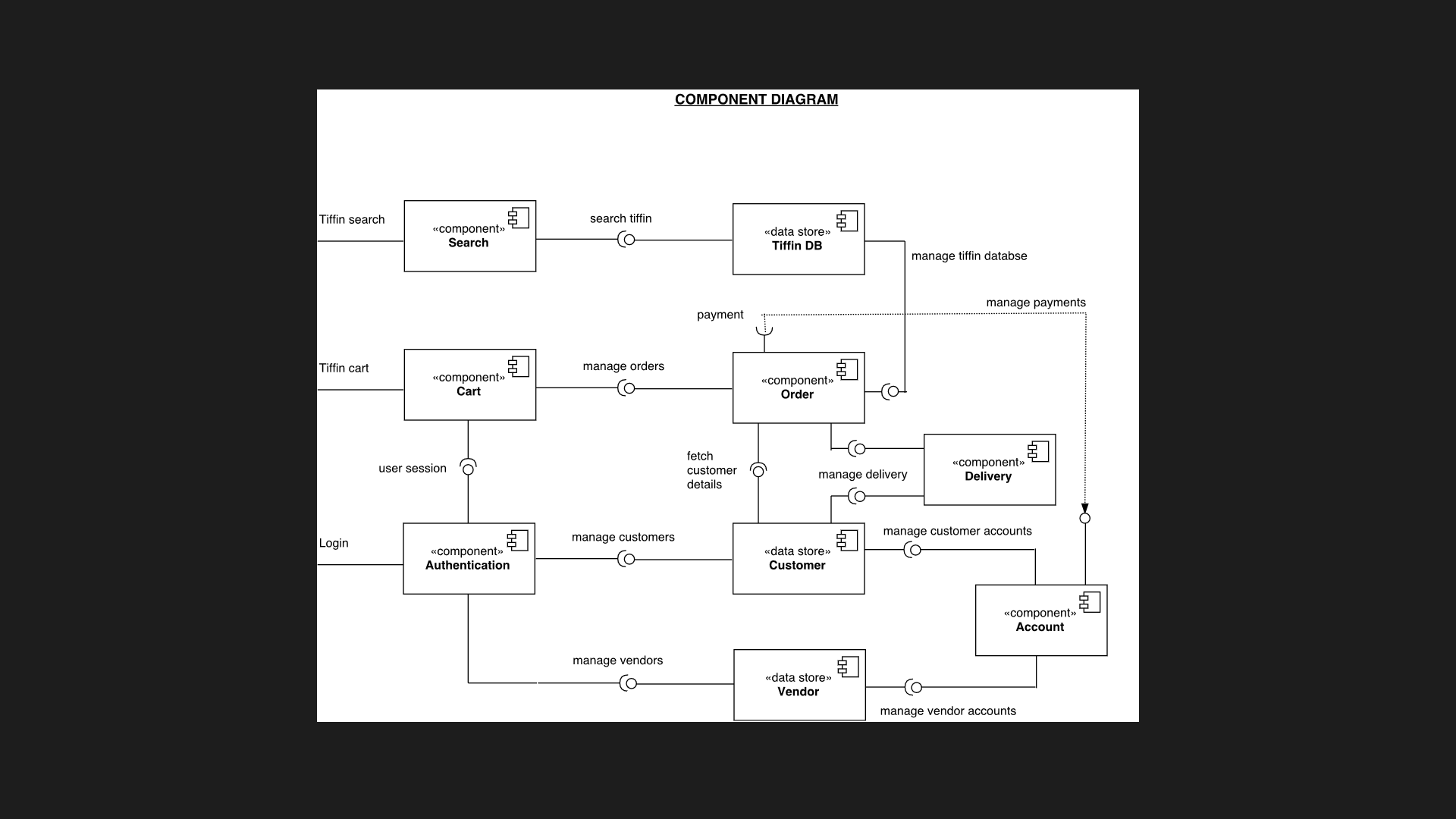
# 5. SYSTEM DESIGN

## Architecture diagram

**Top View**Untitled Diagram (12).png

**Side View**architecutre-diagram2.png

## 5.2 Component Diagram

****

# 6.TESTING

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TEST CASES** | | | | | | | |
|  | | | | | | | |
| **Test Case ID** | **Test Scenario** | **Pre Conditions** | **Test Steps** | **Test Data** | **Expected Results** | **Actual results** | **Pass/Fail** |
|  | | | | | | | |
| **Customer** | | | | | | | |
|  | | | | | | | |
| TU01 | Check customer login with valid data | Tiffin service app should be installed. Login information should exist in the database | 1. Open app and go to login 2.Enter UserID 3.Enter Password 4.Click submit | UserID=lol123 Password=abc | User should login into app | As Expected | Pass |
| TU02 | Check customer login with invalid data | Tiffin service app should be installed. Login information should exist in the database | 1. Open app and go to login 2.Enter UserID 3.Enter Password 4.Click submit | UserID=lol123 Password=abcd | User should not login into app | As Expected | Pass |
| TU03 | Check customer login with null data | Tiffin service app should be installed. Login information should exist in the database | 1. Open app and go to log in 2.Enter nothing in UserID 3.Enter nothing in password 4.Click submit | null | User should not login into app; an error message of null textbox should be shown | As Expected | Pass |
|  | | | | | | | |
| TU04 | Check customer sign up with valid and unique data | Tiffin service app should be installed. | 1. Open app and click on sign up 2.Enter username and email ID 3.Enter password 4.Click sign up | UserName=sample Email=sample@eg.com Password=abcde | User should be assigned the usename and ID to his specific profile, his email should be registered and user should log into app | As Expected | Pass |
| TU05 | Check customer sign up with invalid or non-unique data | Tiffin service app should be installed. | 1. Open app and click on sign up 2.Enter username and email ID 3.Enter password 4.Click sign up | UserName=lol123 Email=sample@eg.com Password=abcde | User should not be assigned the usename and shown an error message that the username is taken | As Expected | Pass |
| TU06 | Check customer sign up with already registered data | Tiffin service app should be installed. | 1. Open app and click on sign up 2.Enter username and already registered email ID 3.Enter password 4.Click sign up | UserName=lol123 Email=sample@eg.com Password=abcde | User should not be assigned the usename and shown an error message that the email ID is registered. | As Expected | Pass |
| TU07 | Check customer sign up with null data | Tiffin service app should be installed. | 1. Open app and click on sign up 2.Enter nothing in username and email ID 3.Enter nothing in password 4.Click sign up | null | User should not be assigned the usename and shown an error message that the fields are empty. | As Expected | Pass |
|  | | | | | | | |
| TU08 | Check the sorting filters with selected different options | Tiffin service app should be installed and customer should be logged in the app | 1.Select an area from the area drop down menu 2.Select from different options of the sorting menus | null | Tiffin options should be seen in a list according to the set filters. | As Expected | Pass |
| TU09 | Check the sorting filters without selecting different options | Tiffin service app should be installed and customer should be logged in the app | Do nothing | null | Show promoted and/or popular tiffins | As Expected | Pass |
| TU10 | Check the selection of area filter by using GPS provided location | Tiffin service app should be installed with location permission and customer should be logged in the app | 1.Select area filter and choose get my location from GPS | null | Sets location provided by GPS from the device. | As Expected | Pass |
|  | | | | | | | |
| TU11 | Check customize button functionality | Tiffin service app should be installed with location permission and customer should be logged in the app | 1.Select a tiffin from the list. 2.Click on the customize my meal button | null | App should navigate to the customize your tiffin section | As Expected | Pass |
| TU12 | Check customize your tiffin section | Tiffin service app should be installed with location permission; customer should be logged in the app; A tiffin should be selected for customization. | 1.Choose quantity of particular items provided in the selected tiffin from the drop down menu 2.Cancel one or two items from the base tiffin if needed 3.Choose sides if needed | null | The page should calculate and show the final price of the tiffin at the bottom as user selects different options | Takes time in calculation if the user takes less than 2 seconds in selecting the options. | Pass |
| TU13 | Check order now button functionality | Tiffin service app should be installed with location permission; customer should be logged in the app; A tiffin should be selected. | 1.After selecting a tiffin/customizing a tiffin click on Order Now button | null | App should navigate to the payment gateway section | As Expected | Pass |
|  | | | | | | | |
| TU14 | Check payment getway functionality | Tiffin service app should be installed with location permission; customer should be logged in the app; A tiffin should be selected. | 1. Choose from any of the different methods of payment | null | Error message should be shown if the time limit or quantity is exceeded for that tiffin; App should navigate to the particular method of payment section | As Expected | Pass |
| TU15 | Check payment getway functionality for cash on delivery | Tiffin service app should be installed; customer should be logged in the app; A tiffin should be selected and the user should be in the payment by COD section. | 1.Fill in the address details or choose from saved addresses. 2.Click on Place your order. | Address Details = 1,asdfg sjgf, nr bsaiov, ihgfaifj, | An error should be shown if the address is not valid.A message of Tiffin ordered should be displayed otherwise. | Slows down the performance of the app if many users are placing order at the same time. Shows error message for invalid address fields. | Fail |
| TU16 | Check payment getway functionality for online payment by using debit/credit card | Tiffin service app should be installed; customer should be logged in the app; A tiffin should be selected and the user should be in the payment by credit/debit cardsection. | 1.Fill in the address details or choose from saved addresses. 2.Fill in the card details. 3.Click on proceed with order. | Address Details = 1,asdfg sjgf, nr bsaiov, ihgfaifj Card number=xxxxxxxxxxxxx CVV number=655 Password/OTP=1345 Captcha=Afjie55 | Error should be shown for any invalid data entry. A message of Tiffin ordered should be displayed otherwise. | As Expected | Pass |
| TU17 | Check payment getway functionality for mobile wallets | Tiffin service app should be installed; customer should be logged in the app; A tiffin should be selected and the user should be in the payment by mobile wallets section. | 1.Fill in the address details or choose from saved addresses. 2.Choose a mobile wallet. 3.Click on the proceed with order. | null | Error should be shown for any invalid data entry; if the balance in the mobile wallet is not enough.A message of Tiffin ordered should be displayed otherwise. | As Expected | Pass |
|  | | | | | | | |
| TU18 | Check hamburger menu functionality | Tiffin service app should be installed and customer should be logged in | 1.Click on the hamburger menu on the top left corner | null | A menu should open with various options | As Expected | Pass |
| TU19 | Check hamburger menu functionality for Profile | Tiffin service app should be installed and customer should be logged in | 1.Click on the hamburger menu on the top left corner 2.Click on Profile option | null | Profile of user should be shown with his personal details. | As Expected | Pass |
| TU20 | Check hamburger menu functionality for History | Tiffin service app should be installed and customer should be logged in | 1.Click on the hamburger menu on the top left corner 2.Click on History option | null | History of user should be shown with his personal details. | As Expected | Pass |
| TU21 | Check hamburger menu functionality for Favourites | Tiffin service app should be installed and customer should be logged in | 1.Click on the hamburger menu on the top left corner 2.Click on Favourites option | null | Favourites of user should be shown. | As Expected | Pass |
| TU22 | Check hamburger menu functionality for Wallet | Tiffin service app should be installed and customer should be logged in | 1.Click on the hamburger menu on the top left corner 2.Click on Wallet option | null | Wallet of user should be shown with his personal details. | As Expected | Pass |
| TU23 | Check hamburger menu functionality Settings | Tiffin service app should be installed and customer should be logged in | 1.Click on the hamburger menu on the top left corner 2.Click on Settings option | null | General Settings of user should be shown. | As Expected | Pass |
|  | | | | | | | |
| **VENDOR** | | | | | | | |
|  | | | | | | | |
| TV01 | Check add tiffin functionality | Tiffin service app should be installed and vendor should be logged in | 1.Click on the add new tiffin button 2.Fill in the required details of the menu 3.Add quantities for the menu items 4.Add Time limit for ordering tiffin 5.Add type of tiffin 6. Click ok button | Menu item1=Daal Menu item2=Bhaat Menu item3=SubziTimeLimit=15:00 Type=Punjabi Dinner TiffinQuantity=50 | Vendor should be able to add another tiffin to his catalogue | As Expected | Pass |
| TV02 | Check edit tiffin functionality | Tiffin service app should be installed and vendor should be logged in;Vendor should have added a tiffin to be edited | 1.Select PreAdded Tiffin to edit 2.Click on edit pencil button 3.Edit the tiffin details as per the requirements 4.Click Save button | Menu item1=Daal Menu item2=Roti Menu item3=PapadTimeLimit=8:00 Type=Gujarati Lunch TiffinQuantity=25 | Vendor's tiffin catalogue must be updated | As Expected | Pass |
| TV03 | Check providing area functionality | Tiffin service app should be installed and vendor should be logged in;Vendor should have added a tiffin to be edited | 1.Select a Tiffin 2.Click on area button 3.Add/remove the tiffin providing area 4.Click Save button | Area=Navarangpura | Vendor's tiffin providing area in his catalogue must be updated | As Expected | Pass |
| TV04 | Check customization constraints of menu items of tiffin functionality | Tiffin service app should be installed and vendor should be logged in;Vendor should have added a tiffin to be edited | 1.Select Tiffin 2.Click on edit pencil button 3.Click on add customization constraints 4.Select a Menu Item to add constraints 5.Add constraints 6.Click Save button | CustConstMenuItem1=yes NewConstraint=Quantity | Vendor's tiffin catalogue must be updated with customization options | As Expected | Pass |
| TV05 | Check view ordered tiffins functionality | Tiffin service app should be installed and vendor should be logged in;Vendor should have added a tiffin to be edited | 1.Click on Ordered Tiffins | null | Vendor should be shown a list of ordered tiffins that he needs to deliver. | As Expected | Pass |

# 7. FUTURE SCOPE

# Packed corporate meals

# Catering for events

# Website for greater reach

# Third party delivery service

# Automated authentication of delivered tiffins

# 8. CONCLUSION

* Various functionalities of the application were analyzed, planned and designed.
* Risks involved with the application have been analyzed and managed.
* A working prototype for the application has been developed.

# APPENDIX

# Prototype

