A

Project Report on

Online Printing Solution For

Users And Vendors

SUBMITTED IN PARTIAL FULFILLMENT

FOR THE AWARD OF

PG DIPLOMA IN ADVANCE COMMPUTING

From C-DAC, ACTS (Bangalore)

Presented by

AKASH KUMAR KANNAUJIYA	**PRN : 190850120006**
AMIT YADAV	**PRN : 190850120012**
KAUSHAL KUMAR SINGH	**PRN : 190850120045**
PARTH GROVER	**PRN : 190850120066**
PRASHANT KUMAR	**PRN : 190850120073**

ACKNOWLEDGEMENT

We take this opportunity to express our gratitude to all those who helped us in various ways and capabilities in understanding this project and devising this project.

We would like to express our gratitude to the staff of ACTS KP for their supervision, advice, guidance encouragement given from the very early stage of this project.

Also, we are thankful to our friends and contemporaries for their cooperation and compliance.

AKASH KUMAR KANNAUJIYA

AMIT YADAV

KAUSHAL KUMAR SINGH

PARTH GROVER

PRASHANT KUMAR

ABSTRACT

Printing often requires a flash drive or any physical storage and which than need to be given to the vendor available in the market for the documents that are required to be printed. However, with GoPrint user does not need any physical storage , instead user can upload the files at GoPrint and select the available vendor nearby to his/her location and print the documents. The User can also choose the mode of payment as GoWallet or directly pay the vendor.

Vendors on the other hand can register on GoPrint and become a GoPrint enabled vendor, which results in attracting more customers. Vendors can collect the amount from Users via GoWallet or any other mode of payment.

1. INTRODUCTION

Printing often requires a flash drive or any physical storage and which than need to be given to the vendor for the documents that are required to be printed. However, with GoPrint user does not need any physical storage , instead user can upload the files at GoPrint and select the available vendor nearby to his/her location.

With GoPrint any vendor can sign up and which will result in attracting more numbers of customers.

1.1 PURPOSE

GOPRINT, aims to reduce the hassle of taking the documents in physical drive and looking for the suitable vendor in the area for printing.

With GoPrint, user can upload the documents which he/she wants to print and then select the suitable vendor for printing. User can also choose the different mode of payment as well.

1.2 SCOPE

We often find it difficult to find the physical storage device or the vendors available nearby for printing out documents. So, it is desirable to have digital solution to provide an optimal solution for this.

The GoPrint helps user digitally to solve the issue of transferring documents on physical drive and later providing it to the vendor.

GoPrint not only helps the customers, But it also helps the Vendors as well by attracting more number of customers.

1. WORKING

1.1. FLOW FROM PRESENTATION

1.1.1. USERS, will reach to the website (url : www.goprint.com). If the user is new he/she can signup on the site.

1.1.2. If the User is registered, He/She can Login to the website with their Login credentials.

1.1.3. If the Login details are valid, user will be directed to the User-Dashboard else User will be prompted to enter the valid login credentials.

1.1.4. Once User is logged in, User will be redirected to User dashboard, which will consist of following:

a. Home
b. Wallet
c. Vendor Nearby
d. Print Prices
e. Edit Profile
f. Logout
g. Order Summary

1.1.5. User will look for vendor nearby based on his zip code, a list of vendors in that zip code will be loaded.

1.1.6. User can select the desired vendor, followed by uploading the documents.

1.1.7. User can now see his order and order summary in the dashboard.

1.1.8. Order summary will comprise of order status, order data, vendor name, and vendor address.

1.2. HOW IT REACHES SERVICE LAYER

I.Login Authentication:

For login authentication we have used NodeJS, Based on the list option selected(User/Vendor), the Username and password are send via get mode to the service layer (NodeJs),for vendor option vendor table is searched and for customer, Customer table is searched. If the username and the password are valid then 1 is returned or else 0 is returned indicating invalid username or password.

II.Vendor Orders:

- **pendingvendorOrders:**

  Execute a native query to fetch the list of orders from vendors where vendorId is the current vendor id and status of order is pending.

- **vendorOrders:**

  Execute a native query to fetch the list of orders from vendors where vendorId is the current vendor from orders table.

- **UploadOrders:**

  After the vendor is selected by the user and placing the order, the orders the vendor can mark the request as completed after the status of the order is marked as completed in the database.

III.Customer Orders:

- **CustomerOrders:**

  Execute a native query to fetch the list of orders from vendors where customerId is the current vendor from orders table.

- **UploadOrders:**

  Based on the pincode entered and the vendor selected the user raise order request with all details for placing the

order, after successful placing of the order the orders table is updated with status as pending.

- **vendorList:**

  Based on the pincode entered the list of vendors available in that pincode is displayed to the user, from which user can select and place an order.

IV.Upload File:

- **uploadFile:**

  Execute a native query to fetch the Multipart file from the client and covnert it into bytes[] and store it into the data and by fileId, filename,etc creating the url to generate the download the file which is being passed back to the client.

1.3. HOW IT REACHES DAO LAYER

1.3.1. Using Spring boot application which is running on port 8080 will listen to this request through the Rest controller which will redirect it to controller with the particular request key.

1.3.2. After redirection the service layer function present for login will be called implicitly by spring boot due to dependency injected in controller class.

1.3.3. Service class uses the interface for queries provided by JPA repository class.

1.3.4. We have an Entity class for these queries named as user details class where we have mapped the class according to database.

1.3.5. @Entity , @Table , @Column are used for proper mapping as defined by spring .

1.3.6. After receiving data from database it will be returned in the form of a json object like this { "email" :"John", "age":30 }

1.3.7. This json object is read by http client services and data from the object will be read and checked with the username and password.

1.3.8. If the username and password matches, user/vendor will be redirected to their dashboard, if it doesn't matches then it will show invalid username or password.

1.3.9. Once user is logged in, user/vendor can use the functions available in the dashboard.

1.4. LEARNINGS FROM PROJECT :

1.4.1. Spring , Spring Boot , Hibernate

1.4.2. How dependency injection works in spring

1.4.3. Usage of Annotations in SpringBoot

1.4.4. Angular architecture , Data Binding , Event Binding

1.4.5. Service Injection in Components in AngularProject

1.4.6. Using HTML 5 , Bootstrap for desingning the Application

1.4.7. Using Postman App for Integration Testing with http Api

1.4.8. Creating Database Using Mysql

1.4.9. Continous Integration using Github