THE URBAN THRIFT

A PROJECT REPORT

submitted by

ANANYA SUNIL(MUT20CS034)

NIKHIL SAVIO ROZARIO(MUT20CS092)

PARVATHY ANIL(MUT20CS095)

SANDRA JOHN(MUT20CS102)

SEBIN KURIAKOSE(MUT20CS129)

to

The APJ Abdul Kalam Technological University in partial fullfilment of the requirements for the award of the Degree

of

Bachelor of Technology

In

Computer Science & Engineering

Department of Computer Science & Engineering

Muthoot Institute of Technology and Science

Varikoli PO, Puthencruz - 682308

JULY 2023

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

ANANYA SUNIL(MUT20CS034)

NIKHIL SAVIO ROZARIO(MUT20CS092)

PARVATHY ANIL(MUT20CS095)

SANDRA JOHN(MUT20CS102)

SEBIN KURIAKOSE(MUT20CS129)

Place:Varikoli

Date:26/06/2023

Muthoot
Institute of Technology & Science

CERTIFICATE

This is to certify that the report entitled **" THE URBAN THRIFT"**, *submitted by* **ANANYA SUNIL, NIKHIL SAVIO ROZARIO, PARVATHY ANIL, SANDRA JOHN, SEBIN KURIAKOSE** *to Muthoot Institute of Technology and Science, Varikoli for the award of the degree of Bachelor of Technology in Computer Science & Engineering is a bonafide record of the project work carried out by her, under our supervision and guidance. The content of the report, in full or parts have not been submitted to any other Institute or University for the award of any other degree or diploma.*

Ms.Sheena K.V	Ms.Sreenu G	Dr.Anand Hareendran S
Project Guide	Project Coordinator	Head of the Department

Place:Varikoli

Date:26/06/2023

ACKNOWLEDGMENT

We are grateful to the almighty who has blessed us with good health, commitment and continuous interest throughout the project work.

We express our sincere thanks to our guide, **Ms. Sheena K.V**, Assistant Professor, Department of Computer Science And Engineering, Muthoot Institute of Technology and Science and **Dr. Anand Hareendran S**, Professor, Head Of the Department, Muthoot Institute of Technology and Science for their guidance and support which were instrumental in all the stages of the project work and without whom the project could not have been accomplished.

In particular, we also wish to express our sincere appreciation to **Dr. Anand Hareendran S**, Head Of the Department, Muthoot Institute of Technology and Science, who was willing to spend his precious time to give some ideas and suggestion towards this project.

We are grateful to our project coordinator **Ms. Sreenu G** Assistant Professor, Department of Computer Science And Engineering, Muthoot Institute of Technology and Science, for her guidance and support.

We would like to thank **Dr. Neelakantan P.C.**, Principal, Muthoot Institute of Technology and Science, Varikoli for providing us all the necessary facilities.

The last but not the least, we extend our sincere thanks to the entire teaching and non-teaching staff of Computer Science And Engineering of Muthoot Institute of Technology and Science for their help and co-operation throughout our project work.

ABSTRACT

The thrifting website serves as a centralized platform for buyers and sellers to engage in the exchange of pre-owned fashion items. The website offers a user-friendly interface with advanced search functionalities, personalized recommendations, and secure transaction processes. By promoting the reuse of products, the website contributes to sustainability efforts and encourages a more eco-conscious lifestyle. Additionally, the website provides an avenue for individuals to generate passive income by selling their unused possessions. Overall, the thrifting website creates a convenient and sustainable solution for fashion enthusiasts, fostering a thriving community focused on affordability, sustainability, and responsible consumption

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

In today's rapidly evolving retail landscape, the world of thrift websites has emerged as a significant and vibrant sector. These online platforms have transformed the way we approach fashion, offering a unique and sustainable shopping experience that resonates with conscious consumers. This introduction aims to provide a professional overview of the power and potential of thrift websites, highlighting their value proposition, market impact, and benefits for both businesses and consumers.

The concept of thrift shopping has evolved beyond its traditional perception, now encompassing a diverse range of high-quality, pre-loved clothing, accessories, and home goods. Thrift websites provide a convenient and accessible channel for individuals to discover unique fashion pieces, express their personal style, and engage in sustainable shopping practices. Through seamless online transactions and user-friendly interfaces, these platforms enable consumers to explore a vast array of pre-owned items from the comfort of their homes, eliminating geographical constraints and fostering a global community of thrift enthusiasts.

1.2 SCOPE AND MOTIVATION

The scope of thrift websites extends beyond the boundaries of traditional retail. These online platforms have revolutionized the concept of thrift shopping by creating a global marketplace for pre-owned fashion items, accessories, and home goods. Through user-friendly interfaces and advanced search functionalities, thrift websites provide a vast selection of curated secondhand products, catering to diverse consumer preferences and styles.

The motivation behind the growing popularity of thrift websites is rooted in several key factors that have reshaped consumer behavior and industry dynamics. Understanding these motivations is crucial to appreciating the transformative power of thrift websites within the retail landscape:

- Sustainability: One of the primary motivations for the rise of thrift websites is the growing concern for sustainability. Consumers are increasingly aware of the environmental impact of the fashion industry, including textile waste, water consumption, and carbon emissions.

- Affordability: Thrift websites have become a go-to destination for those seeking affordable fashion options. As consumer preferences shift towards value-conscious choices, thrift platforms offer an accessible avenue to access high-quality and unique fashion items at significantly lower prices than their new counterparts.

- Individuality and Personal Style: In an era where personal expression and uniqueness are highly valued, thrift websites offer a treasure trove of one-of-a-kind fashion pieces

- Community and Ethical Consumption: Thrift websites foster a sense of community and social connection. Buyers and sellers come together to share their passion for sustainable fashion, engage in conversations, and support each other's endeavors.

CHAPTER 2

PROPOSED WORK

2.1 OBJECTIVE

- Create a centralized platform: Develop a user-friendly website that serves as a centralized platform for college students to buy and sell second-hand items within their campus community.
- Enhance affordability: Provide college students with a cost-effective alternative to purchasing new items by promoting the sale of gently used goods
- Foster sustainability: Promote sustainable practices by encouraging students to participate in the second-hand market. The website should emphasize the environmental benefits of buying and selling pre-owned items, highlighting the importance of reducing waste and promoting a circular economy.
- Encourage community building: Facilitate a sense of community among college students through the website.

2.2 PROBLEM STATEMENT

The objective of this project is to create a customized thrift sale website exclusively for college students. The website aims to facilitate the exchange, sale, and purchase of pre-owned items in a user-friendly and sustainable manner. By promoting the reuse of products, the platform seeks to minimize waste and encourage eco-consciousness. Additionally, the website provides students with an opportunity to generate passive income by selling their unused possessions. Overall, the project aims to create a student-centric thrift sale website that fosters sustainable practices and supports the financial well-being of college students.

2.3 EXISTING SYSTEM AND PROPOSED SOLUTION

2.3.1 EXISTING SYSTEM

The current thrift system in the world is a dynamic and diverse landscape that encompasses various platforms, models, and practices. Thrift, or secondhand, shopping has gained significant popularity in recent years due to increased awareness of sustainability, affordability, and individuality in fashion consumption. Here are some key aspects of the current thrift system:

- Brick-and-Mortar Thrift Shops: Traditional brick-and-mortar thrift shops continue to play a vital role in the thrift system. These physical stores, often operated by charitable organizations, offer a diverse selection of secondhand items at affordable prices.

- Peer-to-Peer Resale Platforms: Peer-to-peer resale platforms provide individuals with the opportunity to directly sell their pre-owned fashion items to interested buyers. These platforms facilitate person-to-person transactions, allowing sellers to set their prices and negotiate deals.

- Thrift Swapping and Clothing Exchanges: Thrift swapping and clothing exchanges promote a sustainable and community-driven approach to thrift. These initiatives encourage individuals to exchange their unwanted clothing items with others, fostering a spirit of sharing and reducing the need for new purchases.

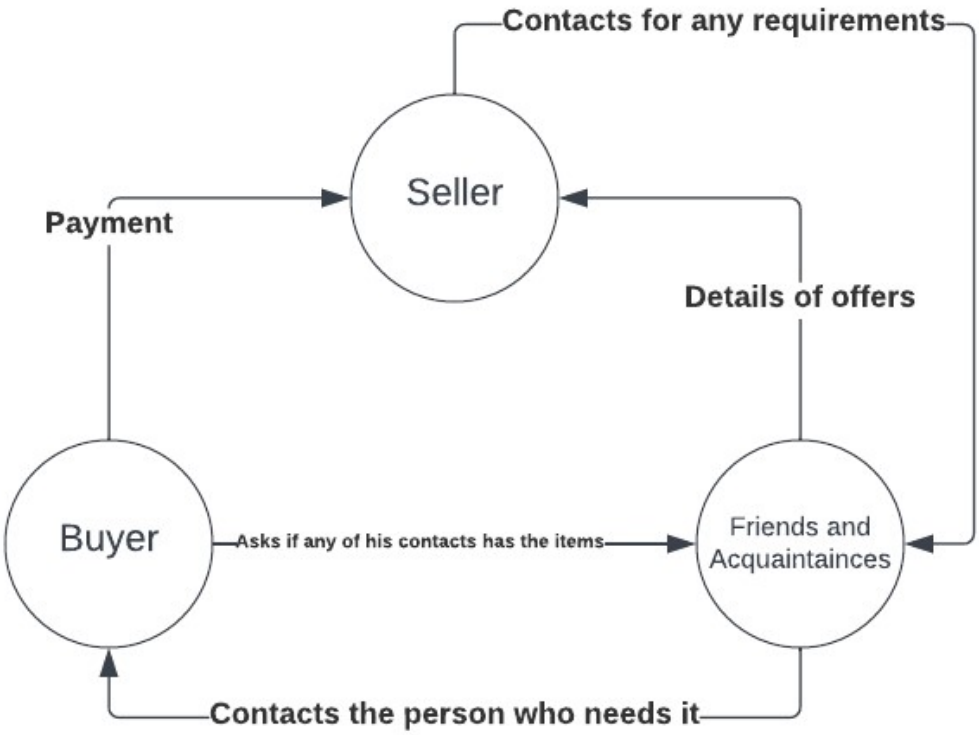

Figure 2.1: CURRENT SYSTEM

2.3.2 PROPOSED SYSTEM

Our platform offers a seamless and sustainable way for students to buy, sell, and exchange pre-owned fashion items. With our user-friendly interface, students can easily navigate and find unique deals. Embracing secondhand fashion not only helps save money but also promotes eco-consciousness and reduces textile waste. Join our student-centric thrifting community and enjoy the benefits of affordability, sustainability, and a vibrant marketplace tailored to your needs.

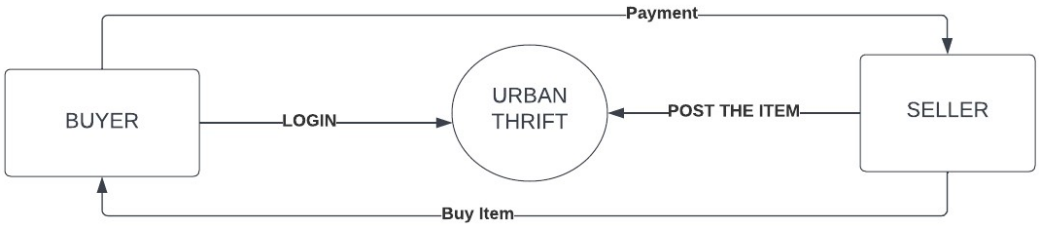

Figure 2.2: PROPOSED SYSTEM

CHAPTER 3

PROJECT DESIGN

3.1 SYSTEM ARCHITECTURE

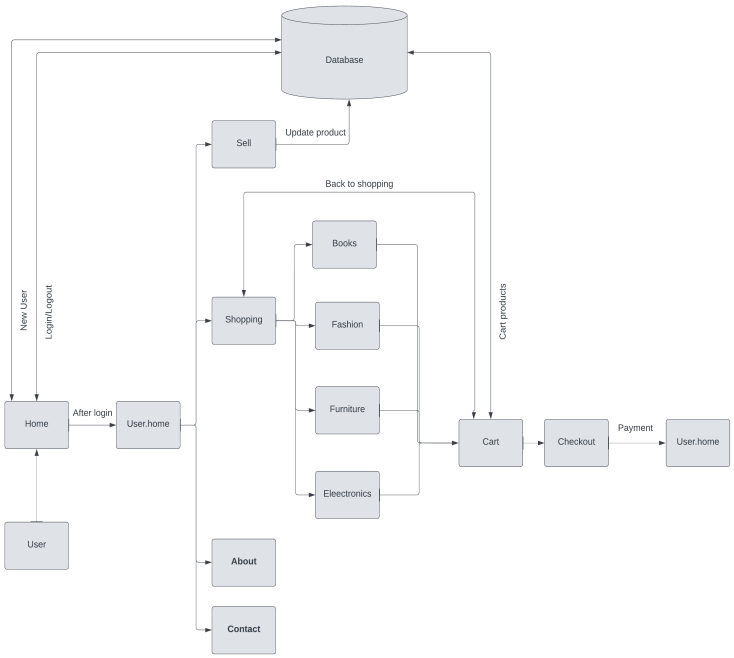

Figure 3.1: SYSTEM ARCHITECTURE

The thrifting website's system architecture consists of several interconnected components that work together to provide a seamless user experience. At the forefront is the User Interface (UI) layer, which users interact with through web pages. This layer allows users to browse items, search, make purchases, manage their profiles, and perform other actions.

Beneath the UI layer lies the Application Layer, which serves as the core logic of the system. It encompasses various services and modules responsible for handling business rules and processing user requests. These services include the Authentication Service, which verifies user credentials and manages secure access to user-specific features.

The Database Layer stores and manages various types of data, including user profiles, item listings, blog posts, contact details, wishlists, and order information. The Application Layer interacts with the databases to retrieve and store the relevant data required for the website's functionality.

Together, these components and layers create a comprehensive system architecture for the thrifting website, enabling users to browse, buy, and sell used items within a college setting. The architecture ensures a seamless flow of data and actions, providing an intuitive and user-friendly experience for all users.

3.2 MODULES

3.2.1 LOGIN MODULE

- User provides username and password.
- Login module verifies credentials with the database.
- User authentication status is provided as a response.
- If successful, an access token is generated for further access to the web app.

3.2.2 MANAGE SHOPPING

- Selected items are displayed in this module.
- Payment details are provided.

3.2.3 MANAGE CART

- Selected items are entered into the cart
- Add and delete options are possible in the cart.

3.2.4 CONTACT

- Contains contact information about the buyers and sellers.

3.3 DATA FLOW DIAGRAM

3.3.1 DFD LEVEL 0

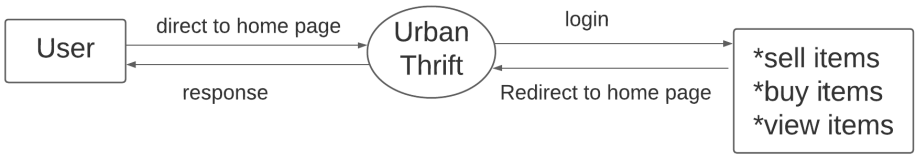

Figure 3.2: DFD LEVEl 0

This Level-0 DFD showcases the user's interaction with the system through the UI, where they can buy and sell second hand items. The system displays the items for thrifting ,items in cart,contact details,payment process etc.

3.3.2 DFD LEVEL 1

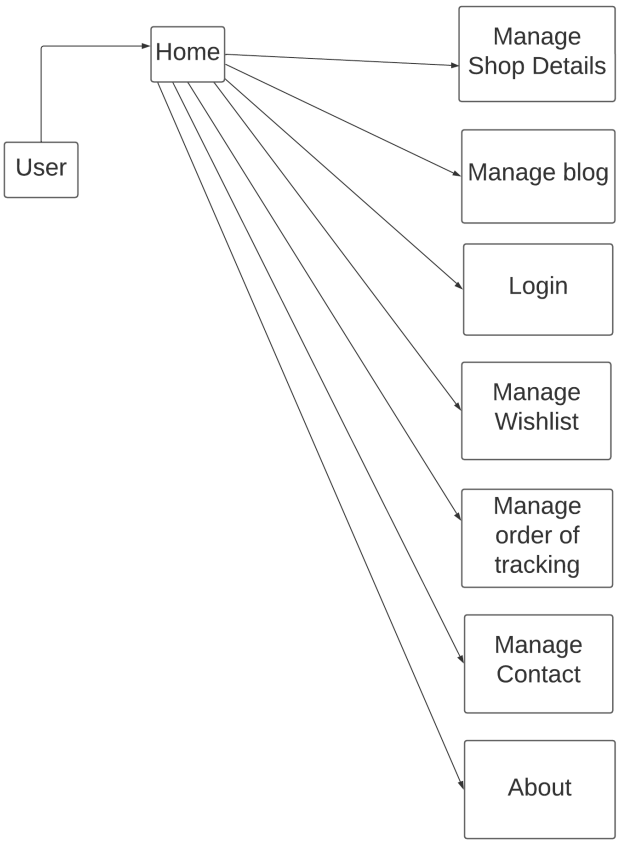

Figure 3.3: DFD LEVEL 1

This Level 1 DFD provides a detailed view of the system flow. The user login module authenticates users and directs them back to home page.The manage shopping module allowsshopping and payment process with data retrieval and storage. The manage cart module handles items in the cart. The contact module verifies contact details and help the buyers to contact with the sellers.

3.3.3 DFD LEVEL 2

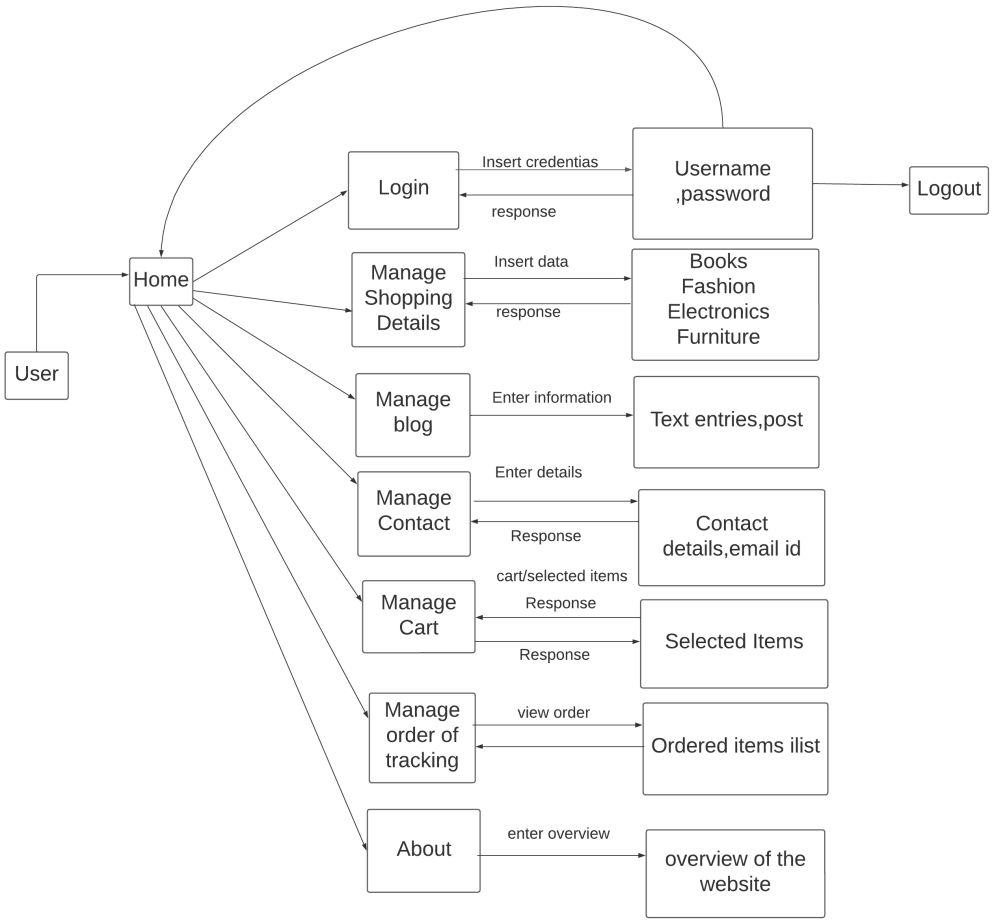

Figure 4: DFD LEVEL2

- Login:

Users are prompted to insert their credentials, including a username and password. The system processes the credentials and provides a response indicating the status of the authentication process.

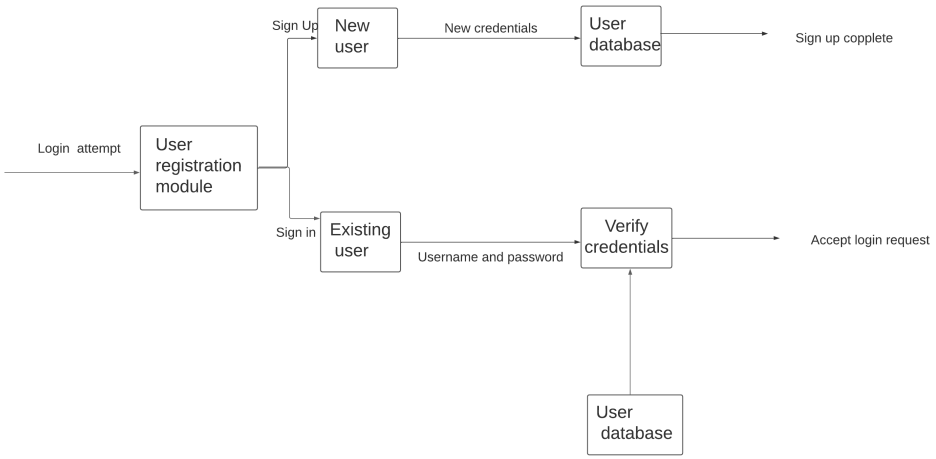

Figure 3.5: LOGIN CONNECTION

- Manage Shopping Details:

Users can insert data related to their shopping details, which might include their preferences, saved addresses, or payment methods. The system processes the data and provides a response indicating the status of the shopping details.

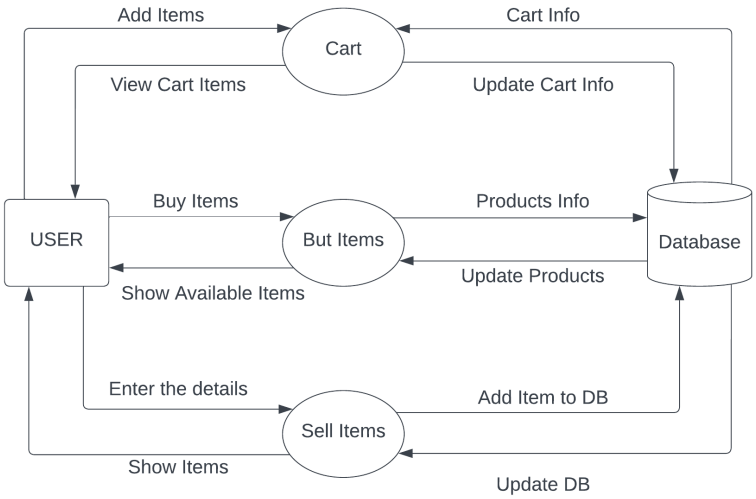

Figure 3.6: SHOPPING MANAGEMENT

- Manage cart Details:

Users can add items to their wishlist or cart, allowing them to keep track of items they are interested in. The system provides a response indicating the status of the selected items.

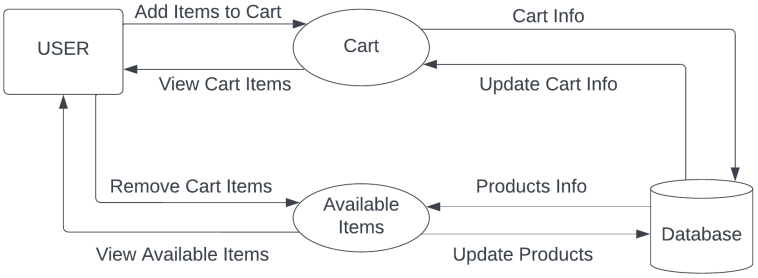

Figure 3.7: CART MANAGEMENT

3.4 DATABASE TABLE DESIGN

3.4.1 USER TABLE

Field Name	Data Type	Constraints
Key	int	PK
Name	varchar	NN
Phone Number	varchar	NN
Password	varchar	NN
Email	varchar	NN

Figure 3.8: USER TABLE

It serves as the foundation, representing users of the system. It includes fields such as key,username, email, password and phone number.

3.4.2 PRODUCT TABLE

Field Name	Data Type	Constraints
Key	int	PK
Name	varchar	NN
Description	varchar	NN
Photo	Image	NN
Category	varchar	NN
Amount	float	NN

Figure 3.9: PRODUCT TABLE

It contains information regarding the product.The table consist of key,name of the product,its description,photo of the product,category and amount.

3.4.3 CART TABLE

Field Name	Data Type	Constraints
Key	int	PK
Product Name	varchar	NN
Amount	float	NN
Quantity	int	NN
Total Amount	float	NN

Figure 3.10: CART TABLE

The cart table contains information related to the items added to the cart.It contains the key, the name of the product, its amount, quantity, and total amount.

3.4.4 COUPON TABLE

Field Name	Data Type	Constraints
Key	int	PK
Code	varchar	NN
Amount	float	NN
Status	varchar	NN

COUPON TABLE

This table contains information related to the payment part. It contains the key of the product, product name, user id, quantity, address of the buyer, date and time of delivery, and total amount.

3.4.5 DATABASE CONNECTION

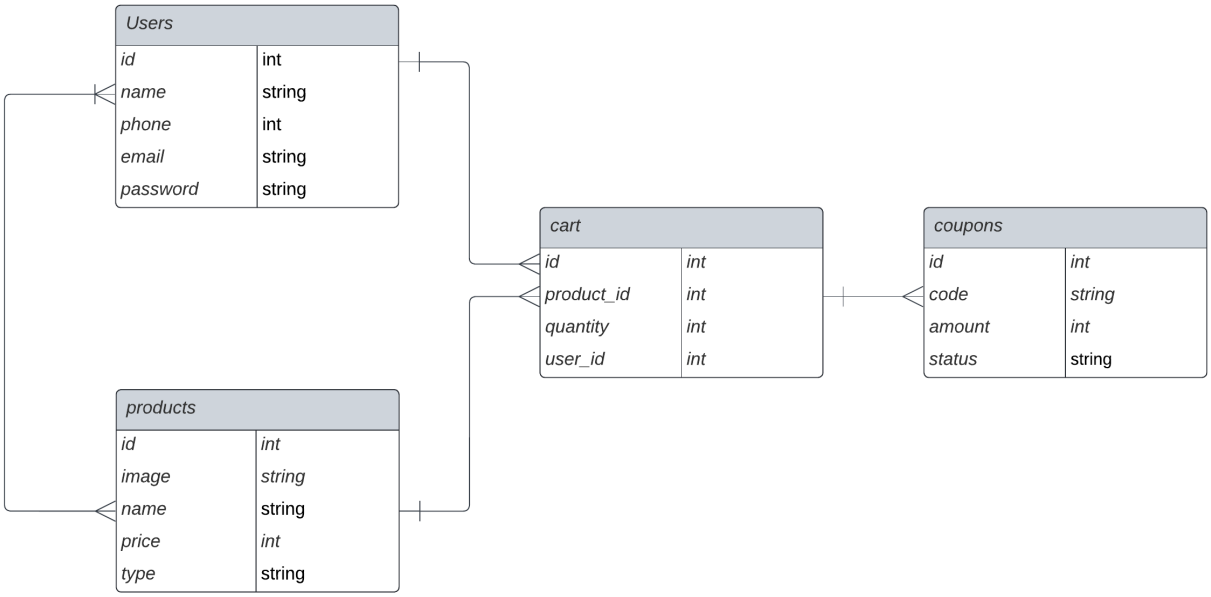

Figure 3.11: DATABASE CONNECTION TABLE

Database Connection Table connects the above tables. User tables, cart tables, product tables, and bill tables are connected to show the flow of data.

3.5 GUI DESIGN

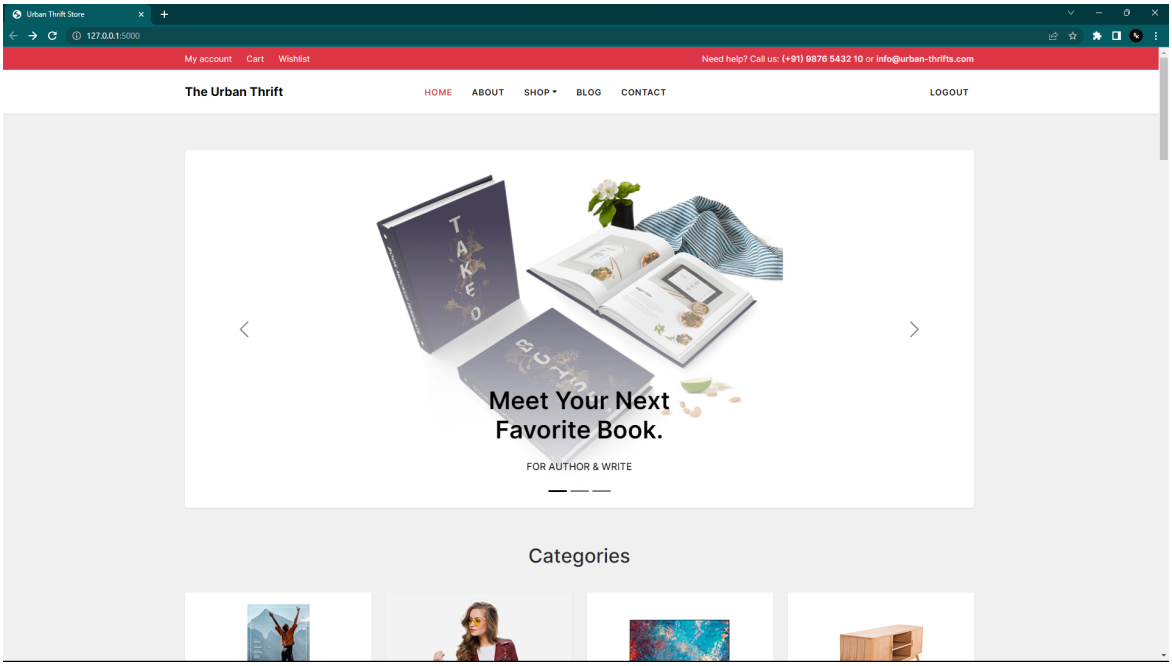

Figure 3.12: GUI DESIGN

- Clean and Modern Layout: Use a clean and modern design approach with a visually appealing layout. Employ a responsive design that adapts well to different screen sizes, including desktops, laptops, tablets, and mobile devices.
- Intuitive Navigation: Ensure easy navigation through the website with a clear and well-organized menu structure. Use recognizable icons or labels for navigation elements and provide breadcrumbs or a search bar for quick access to specific pages or items.
- Categorization and Filtering: Implement a clear categorization system to help users browse items easily. Use dropdown menus or checkboxes to allow users to filter items by categories, such as books, fashion, electronics, or furniture. Include sorting options like price, popularity, or newest listings.
- Search Functionality: Provide a prominent search bar that allows users to search for specific items or keywords. Implement a robust search algorithm

that delivers accurate and relevant results, with autocomplete suggestions and advanced search filters if possible.

- Item Listings: Display item listings in a visually appealing and organized manner. Include essential details like title, description, price, condition, and seller information. Use high-quality images that showcase the items effectively and provide a thumbnail gallery or image carousel for easy browsing.

- User Profiles: Design user profiles with customizable sections for students to showcase their listings, wishlists, reviews, and personal information. Allow users to upload a profile picture and include social media integration for easy sharing.

- Clear Call-to-Action Buttons: Use clear and visually prominent buttons to guide users through actions such as adding items to the cart, making offers, contacting sellers, or completing purchases. Employ appropriate colors and styling to differentiate primary actions from secondary ones.

- Mobile-Friendly Design: Given the prevalence of mobile devices among college students, ensure the website is optimized for mobile use. Utilize responsive design principles to ensure a seamless and engaging experience on smartphones and tablets.

3.6 TECHNOLOGY STACK

Building the thrift website requires a careful selection of technology stacks to ensure optimal performance, scalability, and security. Here are some commonly used technology stacks for developing a thrifting website:

- Front-End Development:

    1. HTML5/CSS3: These foundational technologies are used for creating the structure, layout, and styling of the website.

    2. JavaScript: A versatile programming language for implementing interactive elements, client-side functionalities, and dynamic content.

- Back-End Development:

    1. Programming Language: Python
       Python is a popular and versatile programming language for web development. It offers a wide range of frameworks and tools that simplify the development process and enable the efficient creation of web applications.

    2. Server-Side Framework: Flask
       Flask is a lightweight web framework for Python that simplifies the development of web applications and APIs. It provides routing, request handling, and response generation capabilities

    3. Relational Database Management System (RDBMS): MySQL
       MySQL is a popular relational database management system (RDBMS) that is widely used for web development.It integrates seamlessly with various web development technologies and frameworks, making it a versatile choice for storing and retrieving data in web applications.

- Infrastructure and Deployment:

    1. Web Server: Apache HTTP Server for serving web pages and handling HTTP requests.

2. Version Control: Git for efficient code collaboration, version management, and deployment.

- Additional Components:

1. Payment Integration: Services like Stripe, PayPal, or Braintree for secure payment processing.

2. Image Processing: Libraries such as Cloudinary or ImageMagick for image resizing, optimization, and manipulation.

3.7 SYSTEM REQUIREMENTS

The system requirements for the Thrifting Website are as follows:

3.7.1 HARDWARE REQUIREMENTS

- A server or hosting environment capable of running the chosen technology stack
- Adequate storage capacity to store the system data and any associated files.
- Sufficient processing power to handle concurrent user requests and perform complex computations, especially during seat allocation algorithms.

3.7.2 SOFTWARE REQUIREMENTS

- Operating System: Compatible with the chosen technology stack (e.g., Windows, Linux, macOS).
- Database: SQLite3 database management system to store and manage the system data.
- Programming Languages: Python for server-side and client-side development.

CHAPTER 4

IMPLEMENTATION

4.1 CODE SNIPPETS

4.1.1 COMPONENT ROUTING

```python
from flask import Flask,redirect,url_for
from flask_sqlalchemy   import SQLAlchemy
from flask_login import LoginManager

app = Flask(__name__)
app.config['SECRET_KEY'] = '5791628bb0b13ce0c676dfde280ba245'
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///urban.db'
db = SQLAlchemy(app)
login_manager = LoginManager(app)
login_manager.login_view="login"
login_manager.login_message_category="info"
@login_manager.unauthorized_handler
def unauthorized_callback():
    # Redirect the user to the login page or display an error message
    return redirect(url_for('auth.login'))


# blueprint for auth routes in our app
from .auth import auth as auth_blueprint
app.register_blueprint(auth_blueprint)

# blueprint for non-auth parts of app
from .main import main as main_blueprint
app.register_blueprint(main_blueprint)
```

Figure 4.1: APP ROUTING

4.1.2 STYLING

```css
:root {
    --brand-color: #dc3545;
    --brand-text: #dc3545;
    --color: #282828;
    --brand-shadow: 0px 1px 2px rgba(16, 24, 40, 0.1);
}
body {
    font-family: 'Inter', sans-serif;
    font-size: 16px;
    color: var(--color);
    background-color: #F0F0F1;
}
/* =====[ELEMENTS]===== */
.fz-16 { font-size: 16px!important; }
.brand-shadow { box-shadow: var(--brand-shadow)!important; }


/* =====[FORMS]===== */
.form-check-input { border-color: #98A2B3; }
.form-check-label a { color: var(--brand-color); }
.form-check-input:checked {
    background-color: var(--brand-color);
    border-color: var(--brand-color);
}
.form-check-input:focus {
    border-color: var(--brand-color);
    box-shadow: 0 0 0 .25rem rgba(220, 53, 69, 0.25);
}
.form-control {
    border-color: #D0D5DD;
    box-shadow: 0px 1px 2px rgba(16, 24, 40, 0.1);
    border-radius: 0.25rem;
    padding: .475rem .75rem;
}
.form-control:focus {
    border-color: var(--brand-color);
    box-shadow: 0 0 0 .25rem rgba(220, 53, 69, 0.25);
}

/* =====[BREADCRUMB]===== */
.breadcrumb .breadcrumb-item {
    font-size: 13px;
    text-transform: uppercase;
    letter-spacing: .4px;
}
.breadcrumb .breadcrumb-item a {
    color: var(--color);
    text-decoration: none;
}
```

Figure 4.2: CSS SECTION

4.1.3 BACKEND

```python
from flask import Blueprint
from .models import Products, CartItems, Coupons
from flask import render_template,url_for, request, redirect
from . import db
from flask_login import login_required, current_user

main = Blueprint('main', __name__)

@main.route('/')
@main.route('/home')
def index():

    products = Products.query.all()
    return render_template('index.html',  products=products)


@main.route('/category/<slug>')
def books(slug):
    print(slug)
    products = Products.query.filter_by(type=slug).all()
    print(products)
    return render_template('books.html', products=products)


@main.route('/contact')
def contact():
    return render_template('contact.html')

@main.route('/profile')
def profile():
    return render_template('profile.html')

@main.route('/checkout')
def checkout():
    return render_template('checkout.html')

@main.route('/compelete')
def compelete():
    return render_template('compelete.html')

@main.route('/about')
def about():
    return render_template('about.html')

@main.route('/cart', methods=['POST','GET'])
@login_required
def cart():
    user_id = current_user.id
    cart_items = CartItems.query.filter_by(user_id=user_id).all()
    subtotal = 0
    for i in cart_items:
        subtotal += i.product.price * i.quantity
    tax = 0.18 * subtotal
    if(request.method == 'POST'):
        code = request.form.get('coupon_code')
        coupon = Coupons.query.filter_by(code=code, status='active').first()
        if coupon:
            print('coupon code: ', code, ' Amount: ',coupon.amount)
            subtotal -= coupon.amount
            tax = 0.18 * subtotal
        print('subtotal: ', subtotal, ' tax: ',tax)
        if(subtotal<0):
            subtotal = 0
            tax = 0
        return {'subtotal':subtotal,'tax':tax}
    return render_template('cart.html', cart_items=cart_items, subtotal=subtotal, tax=tax)


@main.route('/add_to_cart', methods=['POST'])
@login_required
def add_to_cart():
    prod_id = request.form.get('prod_id')
    quantity = request.form.get('quantity')

    existing_product = CartItems.query.filter_by(product_id=prod_id, user_id=current_user.id).first()
    if existing_product:
        existing_product.quantity += int(quantity)
        db.session.commit()
    else:
        cart_item = CartItems(product_id=prod_id,quantity=quantity,user_id=current_user.id)
        db.session.add(cart_item)
        db.session.commit()

    return redirect(url_for('main.cart'))

@main.route('/cart_action', methods=['POST'])
@login_required
def cart_action():
    if(request.form.get('delete_item')):
        delete_item_id = request.form.get('prod_id')
        print(' prod id: ',delete_item_id)
        delete_item = CartItems.query.filter_by(product_id=delete_item_id,user_id=current_user.id).first()
        db.session.delete(delete_item)
        db.session.commit()
    elif(request.form.get('quantity')):
        quantity = request.form.get('quantity')
        prod_id = request.form.get('prod_id')
        print('quantity: ',quantity, ' prod id: ',prod_id)
        existing_product = CartItems.query.filter_by(product_id=prod_id, user_id=current_user.id).first()
        print(existing_product)
        if(int(quantity)==0 and existing_product):
            print('some')
            db.session.delete(existing_product)
            db.session.commit()
            return 'Updated'
        if existing_product:
            existing_product.quantity = int(quantity)
            db.session.commit()
    return 'Updated'

@main.route('/checkout', methods=['POST'])
@login_required
def checkout_post():
    CartItems.query.filter_by(user_id=current_user.id).delete()
    db.session.commit()
    return redirect(url_for('main.index'))
```

Figure 4.3: SERVER ROUTING

4.1.4 BACKEND AUTHENTICATION

```python
from flask import Blueprint
from flask import render_template, redirect, url_for, request, flash
from werkzeug.security import generate_password_hash, check_password_hash
from .models import Users
from flask_login import login_user, login_required, logout_user
from . import db

auth = Blueprint('auth', __name__)

@auth.route('/login')
def login():
  return render_template('login.html')

@auth.route('/login', methods=['POST'])
def login_post():
    email = request.form.get('email')
    password = request.form.get('password')
    remember = True if request.form.get('remember') else False

    user = Users.query.filter_by(email=email).first()

    if not user or not check_password_hash(user.password, password):
        flash('Please check your login details and try again.')
        return redirect(url_for('auth.login'))

    login_user(user, remember=remember)
    return redirect(url_for('main.index'))

@auth.route('/register')
def register():
    return render_template('register.html')

@auth.route('/register', methods=['POST'])
def register_post():
    username = request.form.get('username')
    email = request.form.get('email')
    phone = request.form.get('phone')
    password = request.form.get('password')

    user = Users.query.filter_by(email=email).first()

    if user:
      flash('Email address already exists')
      return redirect(url_for('auth.register'))

    new_user = Users(email=email, username=username, password=generate_password_hash(password, method='sha256'),phone=phone)

    # add the new user to the database
    db.session.add(new_user)
    db.session.commit()
    return redirect(url_for('auth.login'))

@auth.route('/logout')
@login_required
def logout():
  logout_user()
  return redirect(url_for('auth.login'))
```

Figure 4.4: AUTHENTICATION

4.1.5 DATABASE

```python
from flask_sqlalchemy import SQLAlchemy
from flask_login import UserMixin
from urban import db, login_manager

@login_manager.user_loader
def load_user(user_id):
    return Users.query.get(int(user_id))

class Products(db.Model):
    id=db.Column(db.Integer(), primary_key=True)
    image= db.Column(db.String(length=30), nullable=False)
    name= db.Column(db.String(length=30), nullable=False)
    price= db.Column(db.Integer(), nullable=False)
    type= db.Column(db.String(length=30), nullable=False)

class Users(UserMixin, db.Model):
    id=db.Column(db.Integer(), primary_key=True)
    username= db.Column(db.String(length=30), nullable=False)
    phone= db.Column(db.String(length=30), nullable=False)
    email= db.Column(db.String(length=30), nullable=False)
    password = db.Column(db.String(100))

class CartItems(db.Model):
    id=db.Column(db.Integer(), primary_key=True)
    product_id = db.Column(db.Integer, db.ForeignKey('products.id'))
    product = db.relationship("Products", backref='product')
    quantity = db.Column(db.Integer())
    user_id = db.Column(db.Integer, db.ForeignKey('users.id'))

class Coupons(db.Model):
    id=db.Column(db.Integer(), primary_key=True)
    code = db.Column(db.String(length=10))
    amount = db.Column(db.Integer())
    status = db.Column(db.String, default='active')

```

Figure 4.5: DATABASE CONNECTION

4.2 SCREENSHOTS

4.2.1 LOGIN PAGE

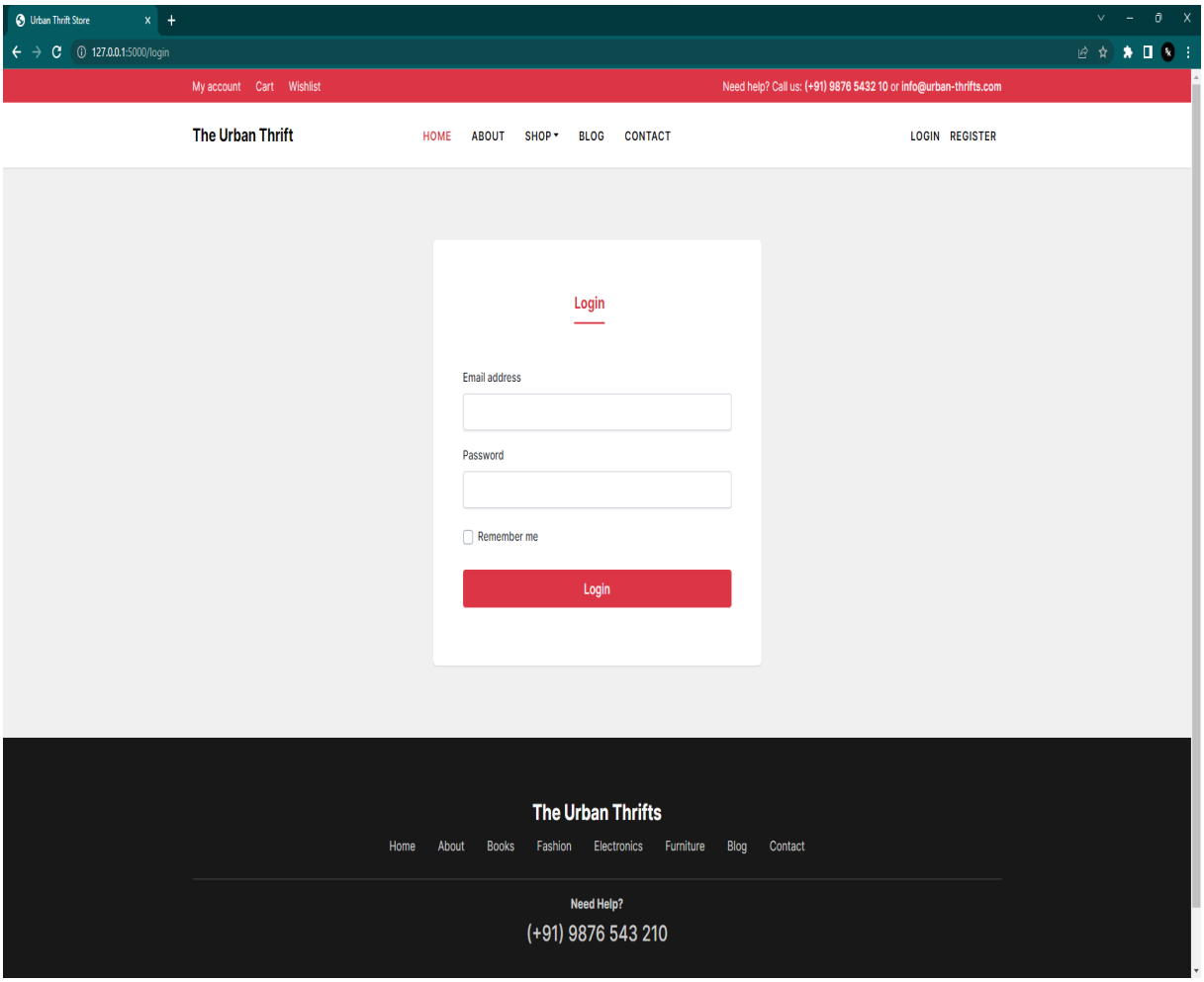

Figure 4.6: LOGIN PAGE

4.2.2 REGISTER PAGE

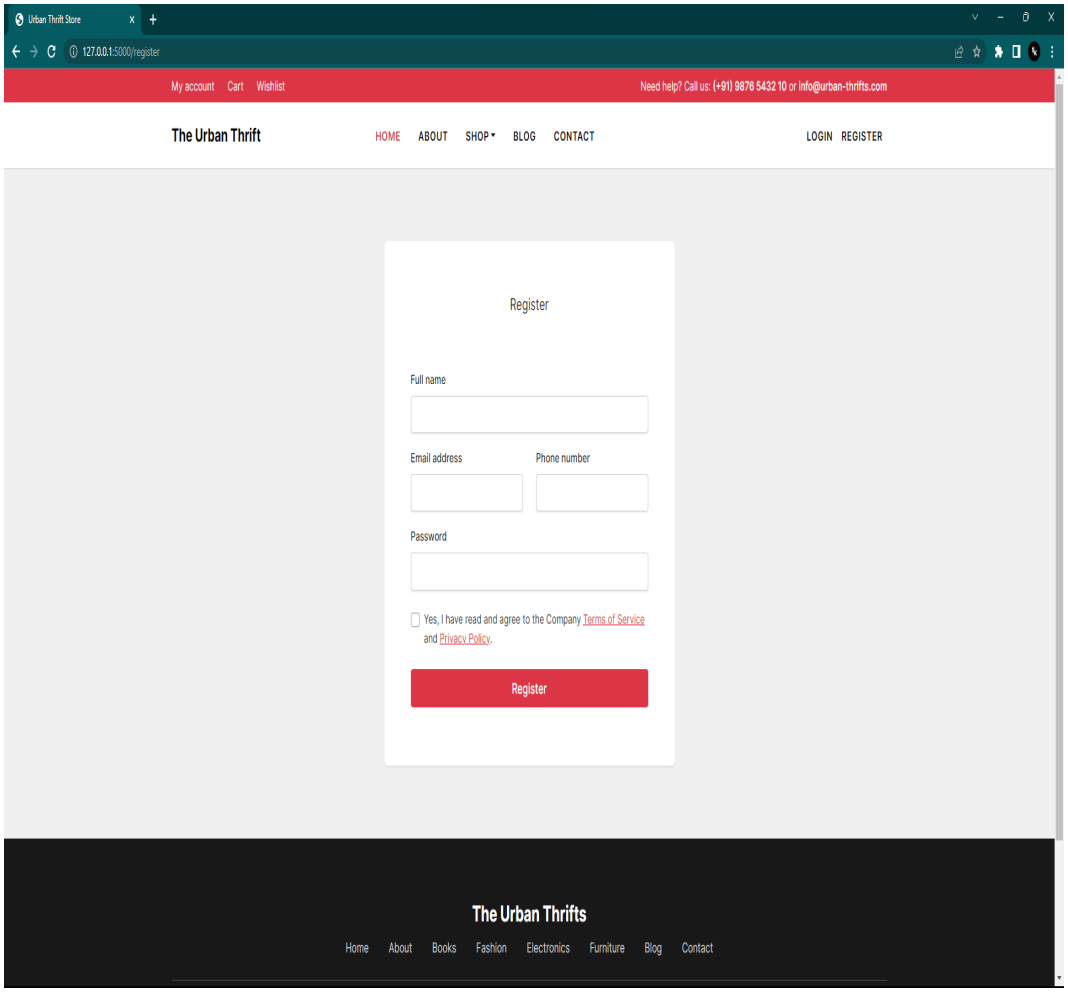

Figure 4.7: REGISTER PAGE

4.2.3 CART PAGE

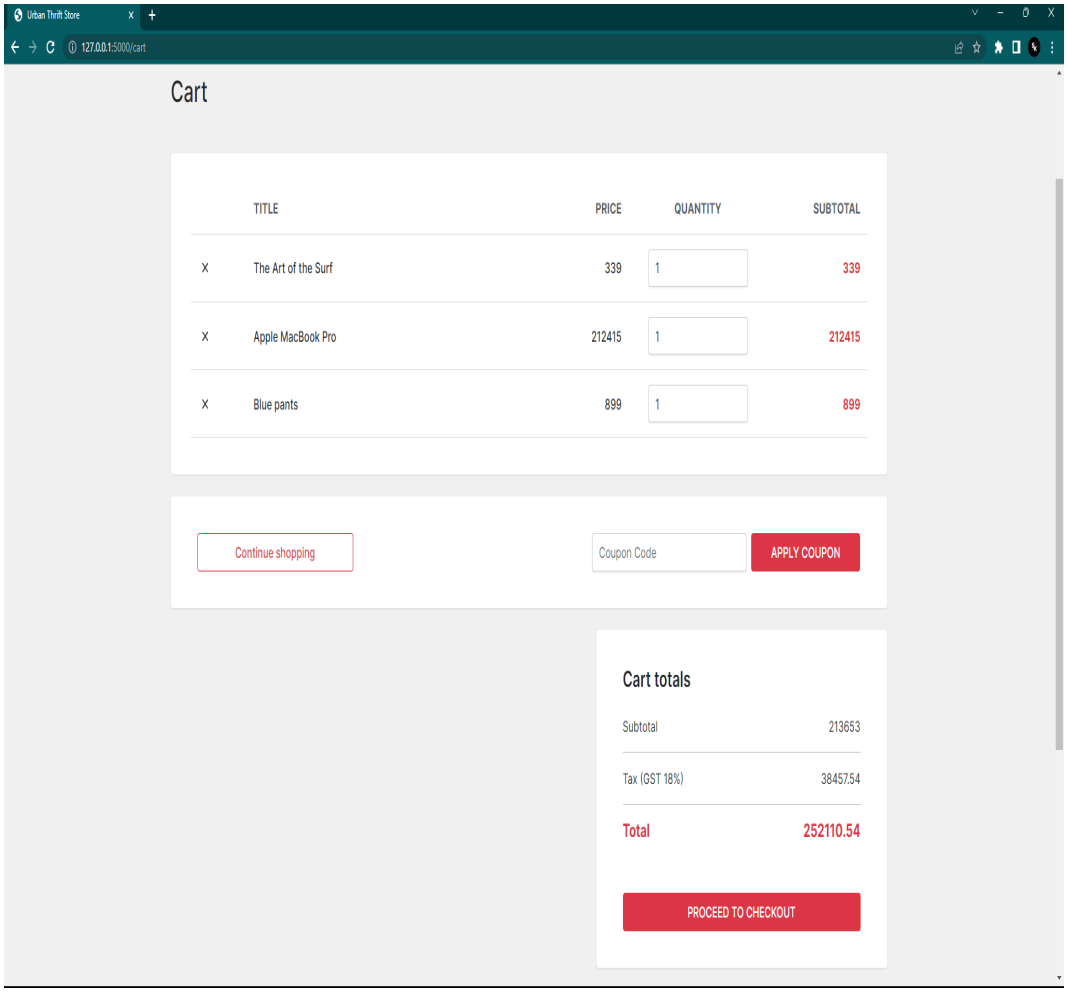

Figure 4.8: CART PAGE

4.2.4 CHECKOUT PAGE

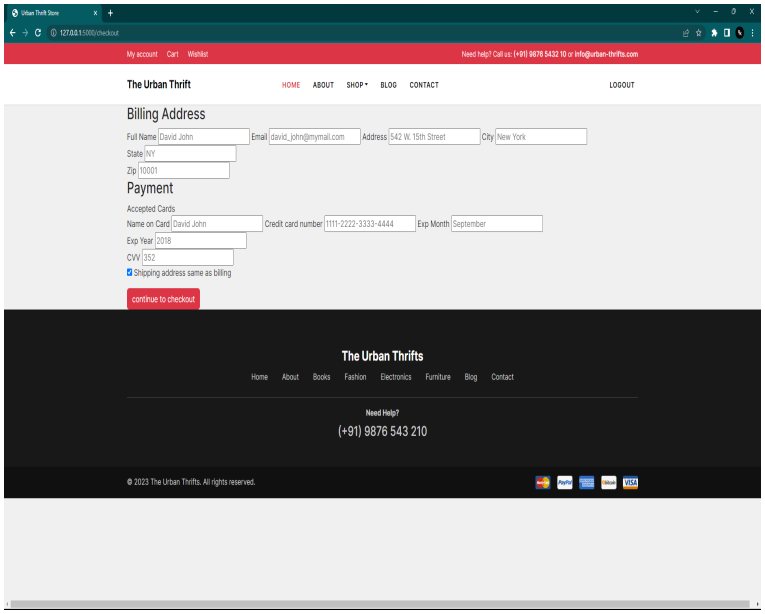

Figure 4.9: CHECKOUT PAGE

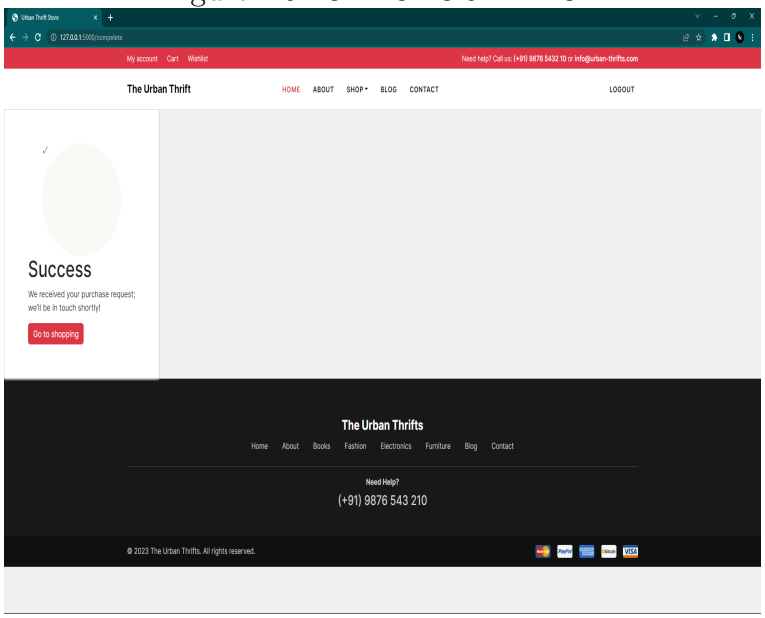

Figure 4.10: SUCCESS PAGE

CHAPTER 5

CONCLUSION

In conclusion, the thrifting website revolutionizes the fashion industry by providing a centralized platform for buying, selling, and exchanging pre-owned fashion items. With its user-friendly interface, advanced search features, and personalized recommendations, the website enhances the shopping experience, making it convenient and enjoyable for users. By promoting the reuse of products, the website significantly contributes to waste reduction and the conservation of resources, aligning with sustainability goals. With secure transactions and a focus on trust, the website ensures a reliable and safe environment for buyers and sellers to engage in transactions, fostering a sense of confidence among users

Overall, the thrifting website empowers individuals to make conscious fashion choices and actively participate in a community dedicated to responsible consumption. By embracing the power of secondhand fashion, the website paves the way for a more sustainable, inclusive, and environmentally conscious future in the fashion industry. Through its transformative approach, the thrifting website not only provides a convenient platform for fashion enthusiasts but also promotes the values of affordability, sustainability, and ethical fashion practices.

5.1 REFERENCES

1. Sabhani A. "The Feasibility of Thrift Store in College Campuses in Kharghar" ,National Institute of Fashion Technology (2017).

2. Claudia E Henninger,Nina Bürklin,Kirsi Niinimäki,"The clothes swapping phenomenon –when consumers become suppliers"Journal of Fashion Marketing and Management

3. Mansfield S. "The impact of digitalization on the thrift store: examining how e-commerce and mobile applications are reshaping the thrift retail market". Journal of Retailing and Consumer Services (2020).

4. Cervellon, Wernerfelt, Merunka. "Preloved Fashion E-commerce: Consumer Motivation and Behaviour in Sustainable Fashion Consumption"(2012) Gupta, "Consumers' Motivations for Online Shopping: An Exploratory Investigation of Indian Students "(2016)

5. Pavlou P.A," Consumer acceptance of electronic commerce: Integrating trust and risk with the technology acceptance model", International Journal of Electronic Commerce(2003)

6. E. Taiebi Javid, M. Nazari, M. R. Ghaeli,"Social media and e-commerce: A scientometrics analysis", International Journal of Data and Network Science(2003)