



Daffodil
International
University

PROGRAM OF STUDY

BSc (Hons) in Computing & Information System (CIS)

Course Code: CIS499

PROJECT TITLE: FOODPATRONS



A dissertation submitted to the Department of Computing and Information System (CIS), Daffodil International University (DIU) for the partial fulfillment the degree of BSc (Hons) in CIS.

Semester: FALL 2021

SUPERVISED BY

Md Sarwar Hossain Mollah
Associate Professor and Head
Department of CIS
Daffodil International University

SUBMITTED BY

Fattahul Islam
ID: 181-16-231
Department of CIS
Daffodil International University

APPROVAL

This Project titled “**Foodpatrons**”, Submitted by **Fattahul Islam** ID No: **181-16-231** to the Department of Computing & Information Systems, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on- 13-02-2022.

BOARD OF EXAMINERS



Mr. Md Sarwar Hossain Mollah
Associate Professor and Head
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Chairman



Ms. Nayeema Rahman
Sr. Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Mr. Abdullah Bin Kasem Bhuiyan
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Mohammad Shorif Uddin
Professor
Department of Computer Science and Engineering
Jahangirnagar University, Savar, Dhaka

External Examiner

Declaration

I hereby declare that; this project has been done by me under supervision of **Ms. Nayeema Rahman, Senior Lecturer**, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By



Mohammad Sarwar Hossain Mollah
Associate Professor and Head
Department of CIS
Daffodil International University

Submitted By



Name: Fattahul Islam
ID: 181-16-231
Department of CIS
Daffodil International University

Acknowledgement

At first, I would like to express my gratitude to almighty Allah for enabling me the strength and opportunity to complete the report within the schedule time successfully.

I like to thank from the core of my heart to the academic supervisor **Md. Sarwar Hossain Mollah**, CIS (Computing & Information System) Department, Daffodil International University, who helped me in every step of my internship. For sure, I am appreciative to all the faculty members of Department of CIS studies of University of Daffodil from whom I have been learning and inspired in different courses.

Finally, I want to thanks my all teachers at Daffodil International University, my friends and well-wishers who supported me and courage me from the very beginning of the project work. I am really grateful to all of them.

Dedication

I like to dedicate this project my parents and sister Tangila Islam. Without my parent's courage, support and inspiration I am nothing but incomplete. I want to dedicate to my sister because, she was my guide and She encouraged me to study computer science.

Abstract

In today's world one culture is influenced by another culture. Especially, last decades platform-based service like food delivery, booking service, dating and entertaining application etc. are becoming widely popular. Foodpatrons will be such an application which will be used for finding random partners for outing, getting food delivery and booking restaurants.

This report presents the Use Interface (UI) and User Experience (UX) design planning. The UI/UX is designed based on some popular platform like food panda, Tinder and pathao of Bangladesh. The UI and UX is designed by using Figma, also used some popular editing tool like photoshop and illustrator. The design will be used for developing prototype and testing the prototype in the market.

Table of Contents

Chapter 1 – Introduction	1
Chapter 2 – Initial Study	1
2.1 Background of the project	1
2.2 Problem Area	1
2.3 Possible solution	1
Chapter 3 – Literature Review	2
Chapter 4 – Methodology	2
4.1 Sections of methodology	2
4.2 Implementation plans	2
Chapter 5 – Planning	3
5.1 Project Plan	3
5.2 Risk Management	3
5.3 Change Management	3
5.4 Quality Management	3
Chapter 6 – Feasibility	4
6.1 All possible type of feasibility	4
6.2 Cost Benefit Analysis	6
6.2 DSDM – good or not for this project -PAQ	7
Chapter 7 – Foundation	7
7.1 Problem Area Identification	7
7.1.1 Interview	8
7.1.2 Observation	8
7.1.3 Questionnaires	9
7.2 Rich Picture	9
7.3 Specific problem area identification and description	10
7.4 Possible Solution	10
7.5 Overall Requirement List	11
7.6 What Technology to be implemented (Client/Web/Standalone)	12
7.7 Recommendations and Justifications	12
Chapter 8 – Exploration	13
8.1 Old System Use Case	13
8.2 Activity Diagram	14
8.3 Full System Use Case	15

8.4 Full System Activity Diagram	16
8.5 Requirement Catalogue	18
8.6 Prioritized Requirement List (PRL).....	21
8.7 Prototype of new system	22
Chapter 9 – Engineering	64
9.1 New System Modules.....	64
9.2 Use Case diagram of the Foodpatrons	65
9.3 Peter Chen ERD Diagram.....	66
9.4 System Interface Hand drawing Design / Prototype	67
9.5 System Interface Design / Prototype	75
Chapter 10 – Deployment / Development	104
10.1 Core Module Coding Samples	104
10.2 Possible problem breaks down.....	108
10.3 Prioritization while developing.....	109
Chapter 11 – Testing.....	110
11.1 Test Plan Acceptance	110
11.2 Test Case	111
11.3 Unit Testing	112
11.4 Module Testing.....	124
11.5 Integration Testing	126
11.6 Security Testing	128
Chapter 12 – Implementation.....	130
12.1 Training	130
12.2 Big Bang	130
12.3 Scaling (is there any plan?)	130
12.4 Load Balancing.....	131
Chapter 13 – Critical Appraisal and Evaluation	131
13.1 Objective that could be met	131
13.2 Objective that totally don't meet of touched	131
Chapter 14 – Lessons Learned.....	132
14.1 Pre project – review – closing.....	132
14.2 Critical appraisal.....	132
14.3 What have I learned	132
14.4 What problem I have faced.....	132

14.5 What solutions occurred	132
Chapter 15 – Conclusion.....	132
15.1 Summary of the project	132
15.2 Goal of the project	133
15.3 Success of the project	133
15.4 What I have done in the documentation (stages / activities / plans)	133
15.5 Value of the project.....	133
15.6 My experience	133
Plagiarism Report	134

List of Figures

- Figure 1: WBS (Work – Breakdown Structure)
- Figure 2: Rich Pictures of Foodpatrons
- Figure 3: Client Server Application Model of Foodpatrons
- Figure 4: Old diagram of Foodpatrons
- Figure 5: Customer Activity Diagram of Foodpatrons
- Figure 6: Customer Activity Diagram of Foodpatrons
- Figure 7: Restaurant Owner Activity Diagram of Foodpatrons
- Figure 8: Customer Activity Diagram of Foodpatrons
- Figure 9: Delivery Man Activity Diagram of Foodpatrons
- Figure 10: Customer app Login option page of Foodpatrons
- Figure 11: Customer app login with phone number page of Foodpatrons
- Figure 12: Customer app OTP page of Foodpatrons
- Figure 13: Customer app Email entry page of Foodpatrons
- Figure 14: Customer app Name entry page of Foodpatrons
- Figure 15: Customer app Birth date page of Foodpatrons
- Figure 16: Customer app Gender selection page of Foodpatrons
- Figure 17: Customer app School name page of Foodpatrons

- Figure 18: Customer app Passion's page of Foodpatrons
- Figure 19: Customer app Add photos page of Foodpatrons
- Figure 20: Customer app profile page of Foodpatrons
- Figure 21: Customer app home page of Foodpatrons
- Figure 22: Customer app Partner selection page of Foodpatrons
- Figure 23: Customer app Chat box page of Foodpatrons
- Figure 24: Customer app Add booking page of Foodpatrons
- Figure 25: Customer app Add booking form page of Foodpatrons
- Figure 26: Customer app Restaurants offers page of Foodpatrons
- Figure 27: Customer app Restaurants Search page of Foodpatrons
- Figure 28: Customer app Restaurant's page of Foodpatrons
- Figure 29: Customer app Restaurant's create review page of Foodpatrons
- Figure 30: Customer app Restaurant's view review page of Foodpatrons
- Figure 31: Customer app Restaurant's booking date form page of Foodpatrons
- Figure 32: Customer app sent invitation page of Foodpatrons
- Figure 33: Customer app Food order form page of Foodpatrons
- Figure 34: Customer app Food order Processing page (1) of Foodpatrons
- Figure 35: Customer app Food order Processing page (2) of Foodpatrons
- Figure 36: Customer app Food order Processing page (3) of Foodpatrons
- Figure 37: Customer app order current location tracking page of Foodpatrons
- Figure 38: Customer app delivery complete page of Foodpatrons
- Figure 39: Restaurant Owner Login option page of Foodpatrons
- Figure 40: Restaurant Owner login with phone number page of Foodpatrons
- Figure 41: Restaurant Owner OTP page of Foodpatrons
- Figure 42: Restaurant Owner Email entry page of Foodpatrons
- Figure 43: Restaurant Owner Name entry page of Foodpatrons
- Figure 44: Restaurant Owner Birth date page of Foodpatrons

Figure 45: Restaurant Owner Home page of Foodpatrons

Figure 46: Restaurant Owner History page of Foodpatrons

Figure 47: Restaurant Owner settings page of Foodpatrons

Figure 48: Restaurant Owner Profile page of Foodpatrons

Figure 49: Restaurant Owner money withdraw option page of Foodpatrons

Figure 50: Delivery man Login option page of Foodpatrons

Figure 51: Delivery man login with phone number page of Foodpatrons

Figure 52: Delivery man OTP page of Foodpatrons

Figure 53: Delivery man Email entry page of Foodpatrons

Figure 54: Delivery man Name entry page of Foodpatrons

Figure 55: Delivery man Birth date page of Foodpatrons

Figure 56: Delivery man home page of Foodpatrons

Figure 57: Delivery man history page of Foodpatrons

Figure 58: Delivery man settings page of Foodpatrons

Figure 59: Delivery man profile page of Foodpatrons

Figure 60: Delivery man withdraw method page of Foodpatrons

Figure 61: Use Case diagram of Foodpatrons

Figure 62: Entity Relationship Diagram of Foodpatrons

Figure 63: Customer app Login option page of Foodpatrons

Figure 64: Customer app login with phone number page of Foodpatrons

Figure 65: Customer app OTP page of Foodpatrons

Figure 66: Customer app Email entry page of Foodpatrons

Figure 67: Customer app Name entry page of Foodpatrons

Figure 67: Customer app Birth date page of Foodpatrons

Figure 68: Customer app Gender selection page of Foodpatrons

Figure 69: Customer app School name page of Foodpatrons

Figure 70: Customer app Passion's page of Foodpatrons

Figure 71: Customer app Add photos page of Foodpatrons

Figure 72: Customer app home page of Foodpatrons

Figure 73: Customer app Partner selection page of Foodpatrons

Figure 74: Customer app Chat box page of Foodpatrons

Figure 75: Customer app Add booking page of Foodpatrons

Figure 76: Customer app Add booking form page of Foodpatrons

Figure 77: Customer app Restaurants offers page of Foodpatrons

Figure 78: Customer app profile page of Foodpatrons

Figure 79: Customer app Restaurants Search page of Foodpatrons

Figure 80: Customer app Restaurant's page of Foodpatrons

Figure 81: Customer app Restaurant's create review page of Foodpatrons

Figure 82: Customer app Restaurant's view review page of Foodpatrons

Figure 83: Customer app Restaurant's booking date form page of Foodpatrons

Figure 84: Customer app sent invitation page of Foodpatrons

Figure 85: Customer app Food order form page of Foodpatrons

Figure 86: Customer app Food order Processing page (1) of Foodpatrons

Figure 87: Customer app Food order Processing page (2) of Foodpatrons

Figure 88: Customer app Food order Processing page (3) of Foodpatrons

Figure 89: Customer app order current location tracking page of Foodpatrons

Figure 90: Customer app delivery complete page of Foodpatrons

Figure 91.1: Unit Testing 1 Test Case

Figure 91.2: Unit Testing 1 Test Case

Figure 91.3: Unit Testing 1 Test Case

Figure 91.4: Unit Testing 1 Test Case

Figure 91.5: Unit Testing 1 Test Case

Figure 91.6: Unit Testing 1 Test Case

Figure 91.7: Unit Testing 1 Test Case

Figure 91.8: Unit Testing 1 Test Case

Figure 91.9: Unit Testing 1 Test Case

Figure 92: Unit Testing 2 Test Case

Figure 93: Module Testing 2 Test Case

Figure 94: Integration Testing 2 Test Case

Figure 95: Security Testing 2 Test Case

Figure 96: Add photos

Figure 97: Create review

Figure 98: Add mobile number

Figure 99: profile change

Figure 100: Update profile

List of Tables

Table 1: Total Cost Estimation for the project

Table 2: Earning estimation for the project

Table 3: Estimated Revenue on a Five-year scale

Table 4: Customer app Requirement Catalogue of Foodpatrons

Table 5: Delivery man app Requirement Catalogue of Foodpatrons

Table 6: Restaurants Owner app Requirement Catalogue of Foodpatrons

Table 7: Login Module of Foodpatrons

Table 8: Restaurant booking and order Module of Foodpatrons

Table 9: Delivery man location Module of Foodpatrons

Table 10: User training

Chapter 1 – Introduction

Foodpatrons is a mobile application which is a platform for food delivery, restaurant booking and finding random partner for having food together. Designing user interface of foodpatrons is the major responsibility of the internship period in Daffodil Computers Limited. In this report, author tries to present the user interface design planning of this project.

Chapter 2 – Initial Study

2.1 Background of the project

Foodpatrons UI/UX design project is the final-year-internship project for academic purpose. The idea of this project was provided by DCL to develop the UI/UX design with a view to getting the visual representation of user requirements.

2.2 Problem Area

In our society people are becoming lonelier because of their busy work schedule. Moreover, at certain age people become more alone if they don't have family. In weekend people wants to enjoy their time visiting good restaurants but in an over populated city like Dhaka, often people could not utilize their weekend like the way they wanted to spend. Because of traffic congestion sometimes they don't get any empty table in their desire restaurant because refreshment opportunity is not available in all areas. This problem is very significant in our modern life because we are following more trends and adapting new cultures rapidly. So, we need a solution which can address these problems

2.3 Possible solution

Foodpatrons is such a platform which can solve all of these mentioned problems in one mobile application. In this platform people who are lonely want to have a good weekend by visiting any restaurant with someone they can choose or invite people randomly in this app. They can also book reservation before going the restaurants. They can also order food and get home-delivery by this app. So Foodpatrons will be develop for people who are struggling with their holiday plan or weekend plan. They can find not only their desire restaurant but also desire food and partner for enjoying the food together.

Chapter 3 – Literature Review

Foodpatrons mobile application UI/UX project was given as an internship project. There is no such application in Bangladesh where user can find a random partner to hang out in a restaurant as well as book reservation and order food. In last five years digital commerce is booming and some food delivery platforms like Foodpanda, Sohoz Food, HungryNaki, Pathao Food etc are playing significant role in this industry. Although social media is playing another for meeting new people and making friends. On the other hands some blind dating application like Tinder playing a significant role for blind dating. In this era of globalization, people are adapting different cultures. Especially young generations are grabbing platform-based entertainment opportunities. So, this demand of this type of platform is raising.

Based on the market demand this project is initiated by DCL and started developing the prototype for market testing.

Chapter 4 – Methodology

4.1 Sections of methodology

The UI/UX design project was developed by two sections.

- Primary Data Analysis
- Secondary Data Analysis

Primary Data: Finalizing User requirements based on the understanding of the client/user needs.

Secondary Data: Analyzing existing food delivery Application, Dating Applications and booking reservation application.

4.2 Implementation plans

Using Tools: Designing the User Interface based on all the requirements and designing with a tool named Figma. Major illustration is designed with photoshop and illustrator.

Chapter 5 – Planning

5.1 Project Plan

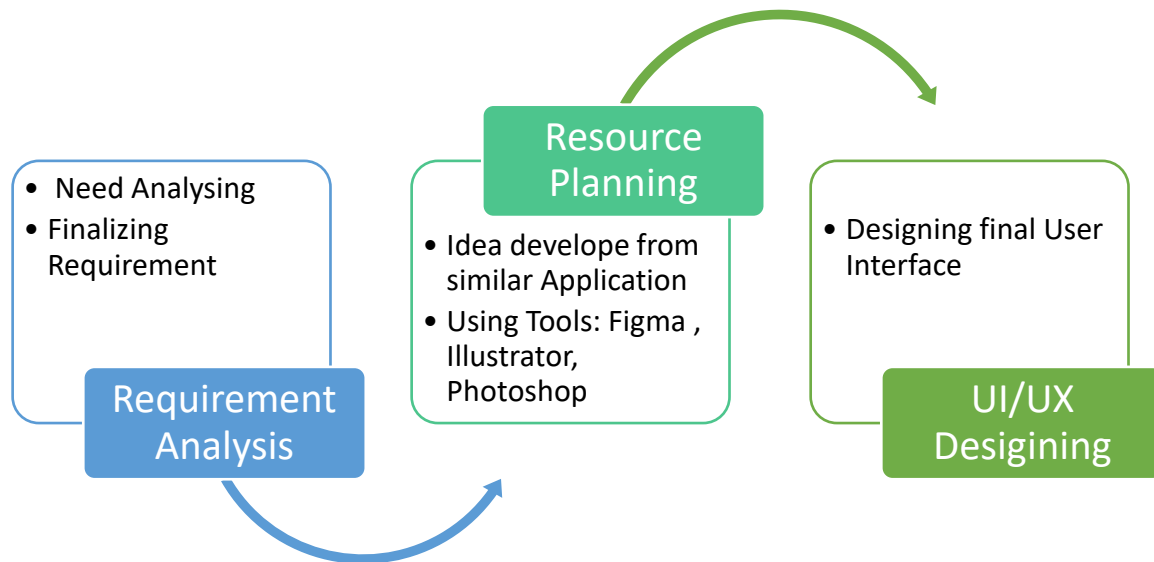


Figure 1: WBS (Work – Breakdown Structure)

5.2 Risk Management

This project may face some social risk. As a conservative country, all member of the society may not accept this kind of dating platform. Moreover, the security of meeting unknown person or going any restaurant with unknown person is risker in a country like Bangladesh. To mitigate this type of social risk we can integrate strict rules and regulation and legal help for this project.

5.3 Change Management

This platform-based project may face some changes in the user requirements. Today's software development process follows Agile methodologies, here changes are warmly acceptable. Any changes in the middle of this project can be adaptable.

5.4 Quality Management

This project will be developed by following agile methodologies and will be tested in every level. All the functional and nonfunctional requirements will be tested before releasing by professional SQA (Software Quality Assurance) Engineers.

Chapter 6 – Feasibility

6.1 All possible type of feasibility

Operational Feasibility:

Because operational feasibility concerns concerning the proposed system's usability, ease-of-use, and needed features, the proposed Foodpatrons system provides adequate validation and verification for user inputs, a really easy and seamless navigation system, a location-based searching option, and a simple request option, all of which can make the system much easier to use. With correct credentials, the admin, restaurant owner, customer, and delivery guy may quickly get access to their app and complete their responsibilities.

Technical Feasibility:

Aspects of technological feasibility include the following:

Foodpatrans users can easily manage their operations, according to a technical feasibility test. The system was built and developed utilizing cutting-edge mobile technology, ensuring that it can work on any Android and ISO phone with a low-bandwidth Internet connection. To ensure safe access to system data, the system contains an access control hierarchy.

As a result, the technical features of this project are explicitly listed below:

Hardware:

- Recommend splitting app and database servers, 2 x 8 CPU servers with 32 GB Ram
- Minimum Server Specification:
 - Processor = Intel (64 bit)
 - Number of Processors = 2
 - Processor core available = 8
 - Memory = 16 Gigabyte
 - OS = Ubuntu 18.04
 - HDD = Up to client
- Minimum Mobile Handset Specification:
 - Android version 7 or above & ios version 7 or above
 - Minimum 1 GB of RAM
 - Data connection

Software used for development:

- DCL ERP for Project management
- Figma for Design
- VS code for Mobile app development

- Pycharm for Back end Development
- Postman for API testing
- Chrome browser for back-end UI testing
- Windows 10 (operating system) & Ubuntu 18.04

Database:

- PostgreSQL & Firebase

Technology:

Back End:

- Python 3.6.8
- NodeJS
- NGINX
- GraphQL
- Pandas

Front End:

- Bootstrap
- Less
- D3.js
- OWL framework

Utilities:

- Google Analytics
- Mandrill
- Github for version controlling

Mobile App:

- Flutter

6.2 Cost Benefit Analysis

The primary goal of a cost-benefit analysis is to determine the earnings and spending assumptions. The entire cost and earnings are compared to determine the benefit. The following is a cost-benefit analysis for this project:

Total cost:

SL no.	Equipment	Year 1	Year 2	Year 3	Year 4	Year 5	Total
1	Mobile Application cost	₹1,05,000	-	-	-	-	₹1,05,000
2	Domain and hosting Cost	₹20,000	₹20,000	₹20,000	₹20,000	₹20,000	₹1,00,000
3	Employee Expenses	₹40,000	₹40,000	₹40,000	₹40,000	₹40,000	₹2,00,000
4	Other Cost	₹20,000	₹20,000	₹20,000	₹20,000	₹20,000	₹1,00,000
	Total Cost	₹1,85,000	₹80,000	₹80,000	₹80,000	₹80,000	₹7,22,000

Table 1: Total Cost Estimation for the project

Total Earn:

SL no.	Earn Sector	Year 1	Year 2	Year 3	Year 4	Year 5	Total
1	Govt. Tax 1	₹1,50,000	₹1,80,000	₹2,10,000	₹2,40,000	₹2,70,000	₹1,050,000
2	Govt. Tax 2	₹2,00,000	₹2,10,000	₹2,40,000	₹2,60,000	₹3,00,000	₹1,210,000
	Total	₹3,50,000	₹3,90,000	₹4,50,000	₹5,00,000	₹5,70,000	₹2,260,000

Table 2: Earning estimation for the project

Total Revenue:

SL no.	Sectors	Year 1	Year 2	Year 3	Year 4	Year 5	Total
1	Total Earning	₹3,50,000	₹3,90,000	₹4,50,000	₹5,00,000	₹5,70,000	₹2,260,000
2	Total Equipment Cost	₹1,85,000	₹80,000	₹80,000	₹80,000	₹80,000	₹7,22,000
	Total Revenue	₹1,65,000	₹3,10,000	₹3,70,000	₹4,20,000	₹4,90,000	₹1,538,000

Table 3: Estimated Revenue on a Five-year scale

The revenue will be increased per year. So, this project will be beneficial for government and also humanly for human.

6.2 DSDM – good or not for this project -PAQ

This project was supplied by DCL, it was a project for Archaic Global Ltd. Which is a copyright protected product. The project has been developed and get live within a very tight scheduled timeframe which includes all the necessary business logic to go live. Though it is a social media app so it may need further modifications during live operations, hence it should be developed iteratively. The iterative development process is ensured by DSDM, which imposes tight rules and constraints. It guarantees that the task is completed in a timely manner. It also guarantees that users are involved throughout the project's duration. As a result, it is evident that DSDM is the optimal technique for Foodpatrons to use.

Chapter 7 – Foundation

7.1 Problem Area Identification

In our society people are becoming lonelier because of their busy work schedule. Moreover, at certain age people become more alone if they don't have family. In

weekend people want to enjoy their time visiting good restaurants but in an overpopulated city like Dhaka, often people could not utilize their weekend like the way they wanted to spend. Because of traffic congestion sometimes they don't get any empty table in their desired restaurant because refreshment opportunity is not available in all areas. This problem is very significant in our modern life because we are following more trends and adapting new cultures rapidly. So, we need a solution which can address these problems

7.1.1 Interview

If you want to acquire information, figure out what the problem is, or figure out what the aim is, an interview is the ideal solution. The genuine remedy might be perfected by locating the problem. As a result, for the proposed system, several questions from the user have been established to take the interview, which are listed below:

1. For Customer:
 - a. People are become loners
 - b. People are become so busy to can't do timely maintain their eating.
 - c. This covid situation people cannot go restaurants like before
2. For Restaurants Owner:
 - a. Restaurants can sell food online
 - b. Restaurants can maintain reservation
3. For Delivery man:
 - a. Observance of Food Quality Standards
 - b. There is no shelter to protect from the rain and sun.
 - c. Charging their phone

7.1.2 Observation

Observations are also useful for gathering user needs and determining the true problem. Observation tactics are mostly used to go to the workplace and figure out how to accept the users. The following are the major factors to consider:

- For customer, they can choose their partner to go on a date
- For customer, they can order food at home
- For customer, they can book a restaurant on the day they want to go
- For customer, they can do anonymous order
- For restaurants owner, they can sell food online
- For restaurants owner, they can offer in food online
- For restaurants owner, they can manage reservation

- For Delivery man, will be allowed to begin his duty.

7.1.3 Questionnaires

Questionnaires are one of the most effective methods for gathering data and asking questions of users and other stakeholders. The user will be asked certain questions, which may be MCQs or short questions. As a result, the following questions are provided:

Questions for Problem Identification	
Questions-1	How can I find a good partner to go on a date?
Ans	
Questions-2	How can I do order food at home?
Ans	
Questions-3	How can I book a restaurant on the day I want to go?
Ans	
Questions-4	How can I do anonymous order?
Ans	
Questions-5	How can I sell food online?
Ans	
Questions-6	How can I share my restaurants offers in online?
Ans	
Questions-7	How can I manage book reservation?
Ans	

7.2 Rich Picture

A rich picture is a top-down or bird's-eye perspective of a system's users' actions. It also depicts the challenges that are at odds, as well as communication and business procedures. The following is a detailed image of the Foodpatrons:

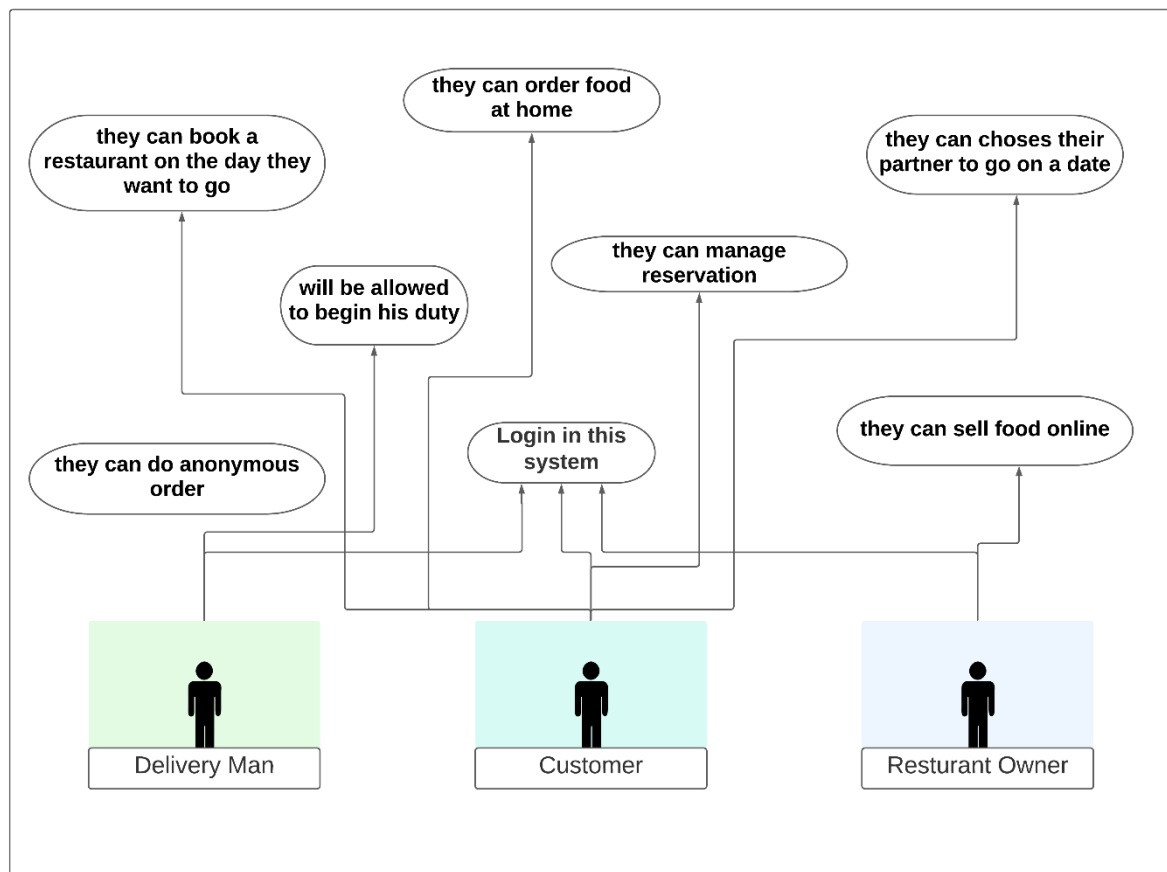


Figure 2: Rich Pictures of Foodpatrons

7.3 Specific problem area identification and description

In the problem area identification portion, a variety of issues are discovered utilizing information collecting approaches. The Food patrons' whole system processes are depicted in a colorful visual. By examining them, I was able to identify some particular issues that need to be addressed

- There is No dating app are available like this
- People are become loners
- People are become so busy to can't do timely maintain their eating.
- This covid situation people cannot go restaurants like before
- Restaurants can sell food online
- Restaurants can maintain reservation
- Observance of Food Quality Standards

7.4 Possible Solution

The recommended remedies to the highlighted issues are as follows:

- Customer can see Restaurants those are close to them

- Customer can choose their partner as they want to
- Customer can book Restaurants according to their date
- Customer can chat with their partner through this app
- Customer can send invitation to their friends to go on a date

7.5 Overall Requirement List

The full list of prerequisites may be seen below:

Functional Requirement

- Create an account
- Manage their account
- Select an item from the menu.
- Customize options for a selected item.
- Customer can see Restaurants those are close to them
- Customer can choose their partner as they want to
- Customer can book Restaurants according to their date
- Customer can chat with their partner through this app
- Customer can send invitation to their friends to go on a date
- Customer can see restaurants offers present time
- Customer can create invitation
- Customer can review a restaurants
- Customer can search restaurants
- Customer can place an order
- Restaurants Owner will manage food orders.
- Restaurants Owner will manage book reservation.
- Restaurants Owner will manage food orders.
- Restaurants Owner can withdraw money in Bikash, Nogod, Visa card, Mastercard
- The delivery lad will be allowed to begin his duty.
- The delivery boy will be able to choose between Online/Offline/Break modes.
- The delivery boy will be able to obtain feedback from the consumer.
- The delivery boy can withdraw money in Bikash, Nogod, Visa card, Mastercard

Non-functional Requirements

- The system should be able to run the program on a variety of platforms, including different versions of the operating system and mobile devices.

- Usability (Efficiency of use, Intuitiveness, Low perceived workload)
- The system must be able to do authentication and payment through bankcard.
- The user will be able to pay using a secure bank payment method.
- The system must be capable of encrypting and decrypting data for the user's login password.

7.6 What Technology to be implemented (Client/Web/Standalone)

The suggested system can be implemented using a variety of technological alternatives. It is critical to select the appropriate technology in order to achieve success. Alternatives include:

Client-Server Application Technology

This is a desktop-based technology, which requires the client's device to be installed and the data to be kept on a distant server. The client program sent a request for data to the distant server, and the server answered.

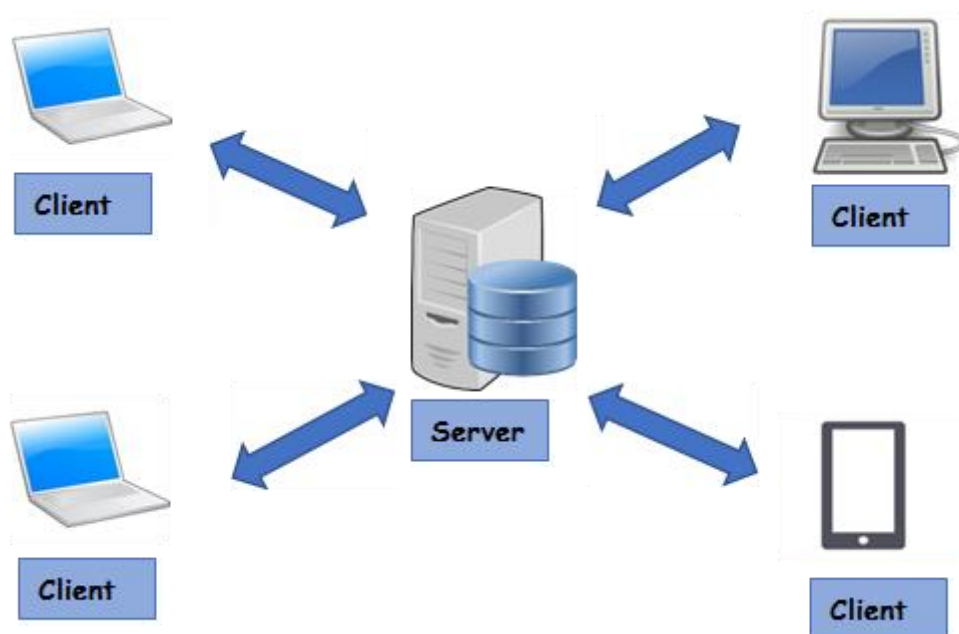


Figure 3: Client Server Application Model of Foodpatrons

7.7 Recommendations and Justifications

The suggested Foodpatrons must be reachable at all times from wherever in the country. Even internet-enabled devices such as smartphones and tablets should be

able to access the system. It should also be able to handle a large number of users at the same time.

Chapter 8 – Exploration

8.1 Old System Use Case

A use case diagram depicts the flow of a system's business operations. I'm going to use the Foodpatrons use case as an example of an older system.

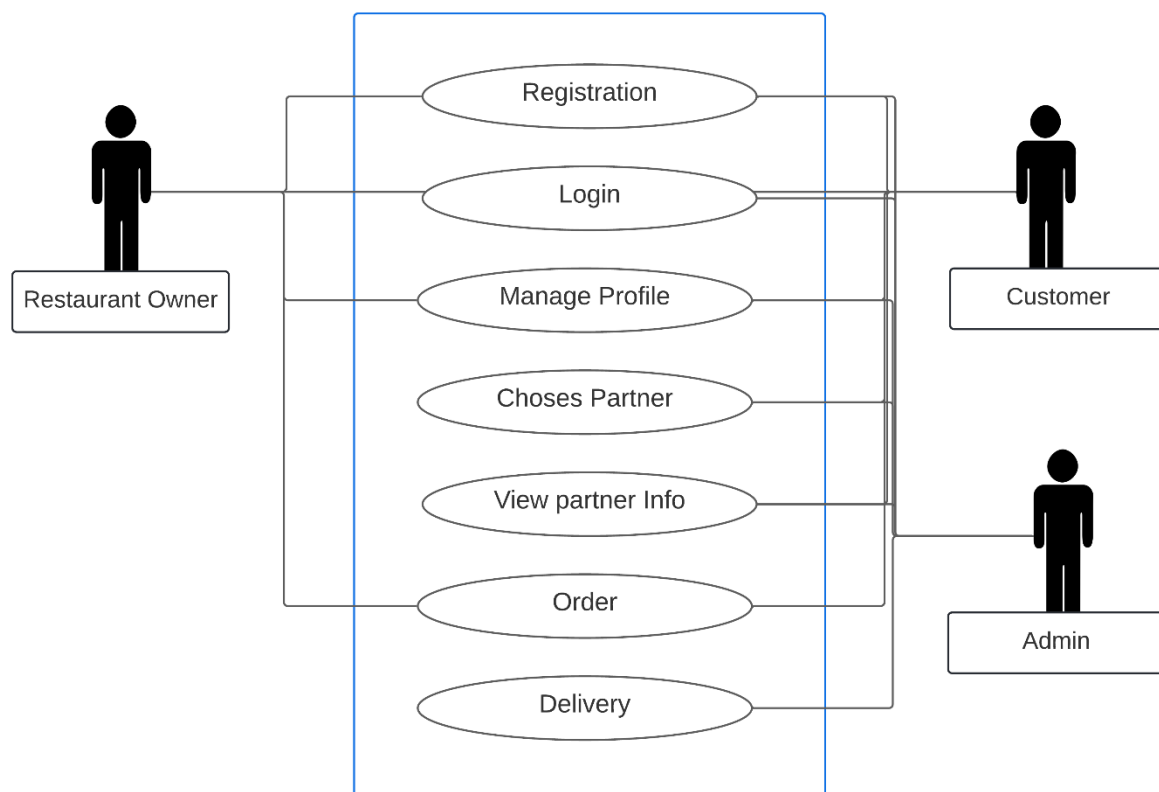


Figure 4: Old diagram of Foodpatrons

8.2 Activity Diagram

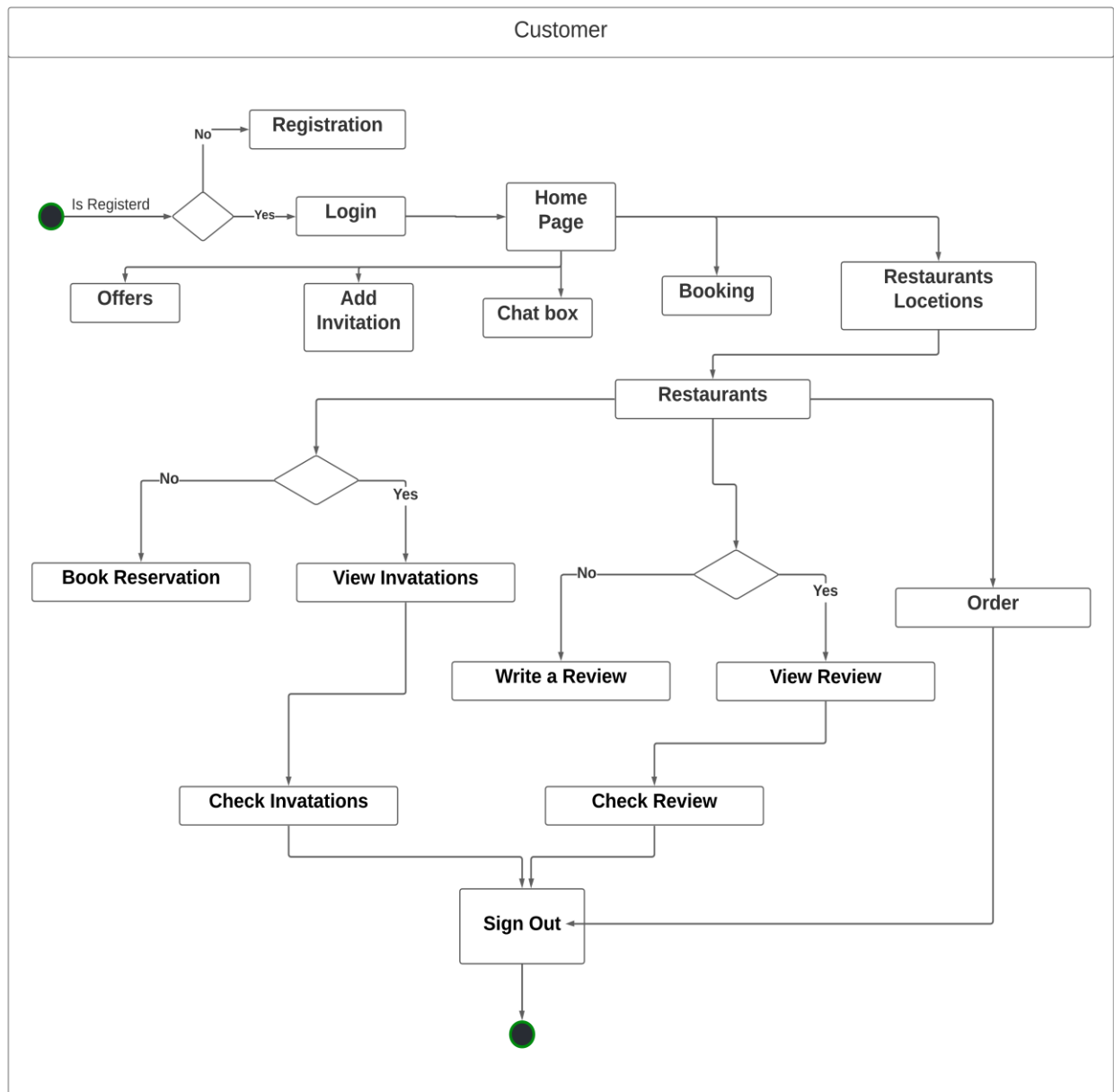


Figure 5: Customer Activity Diagram of Foodpatrons

8.3 Full System Use Case

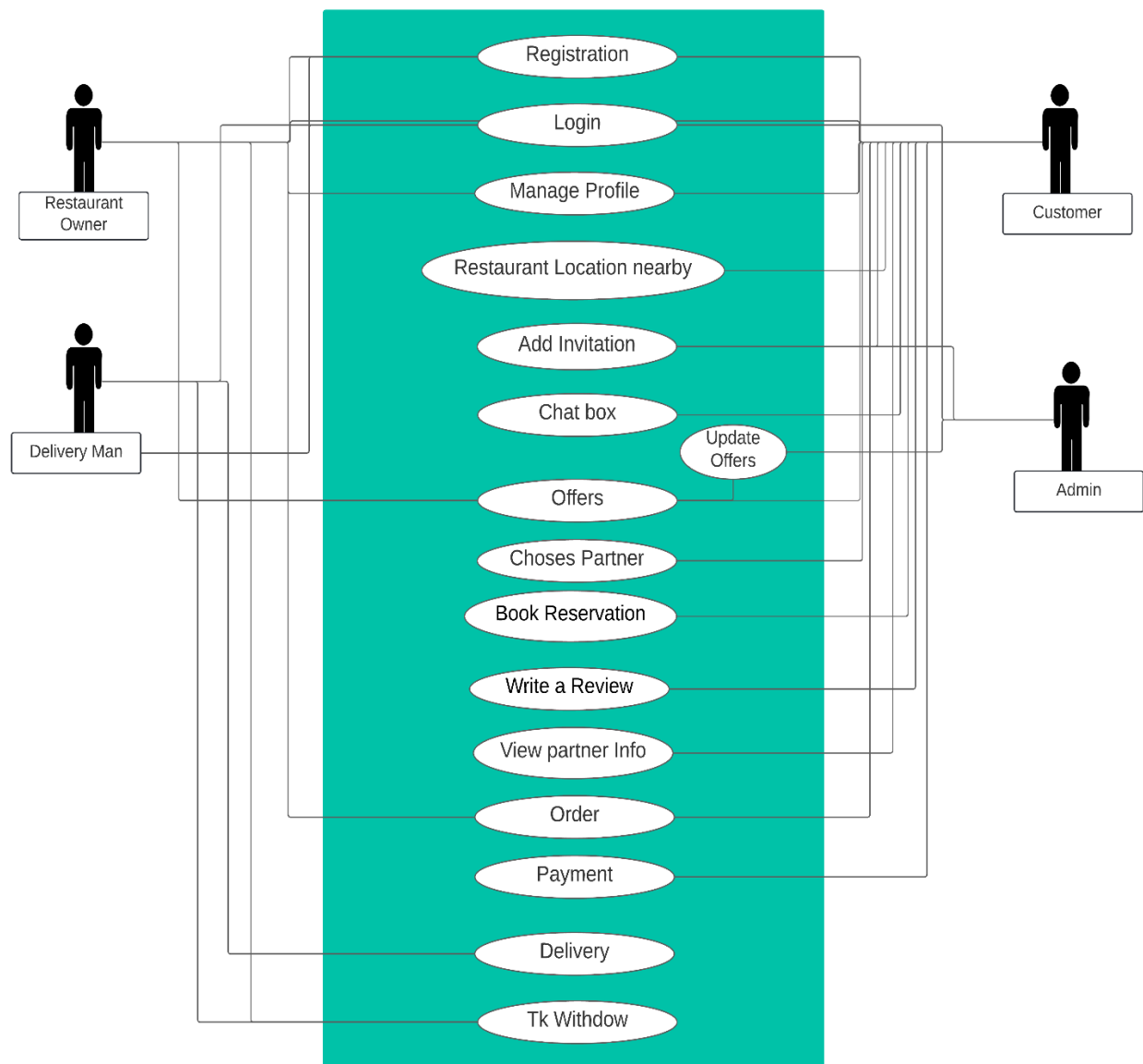


Figure 6: Customer Activity Diagram of Foodpatrons

8.4 Full System Activity Diagram

Users in the proposed Foodpatrons system come in a variety of shapes and sizes, each with their own workflow. Below are the activity diagrams for various work flows:

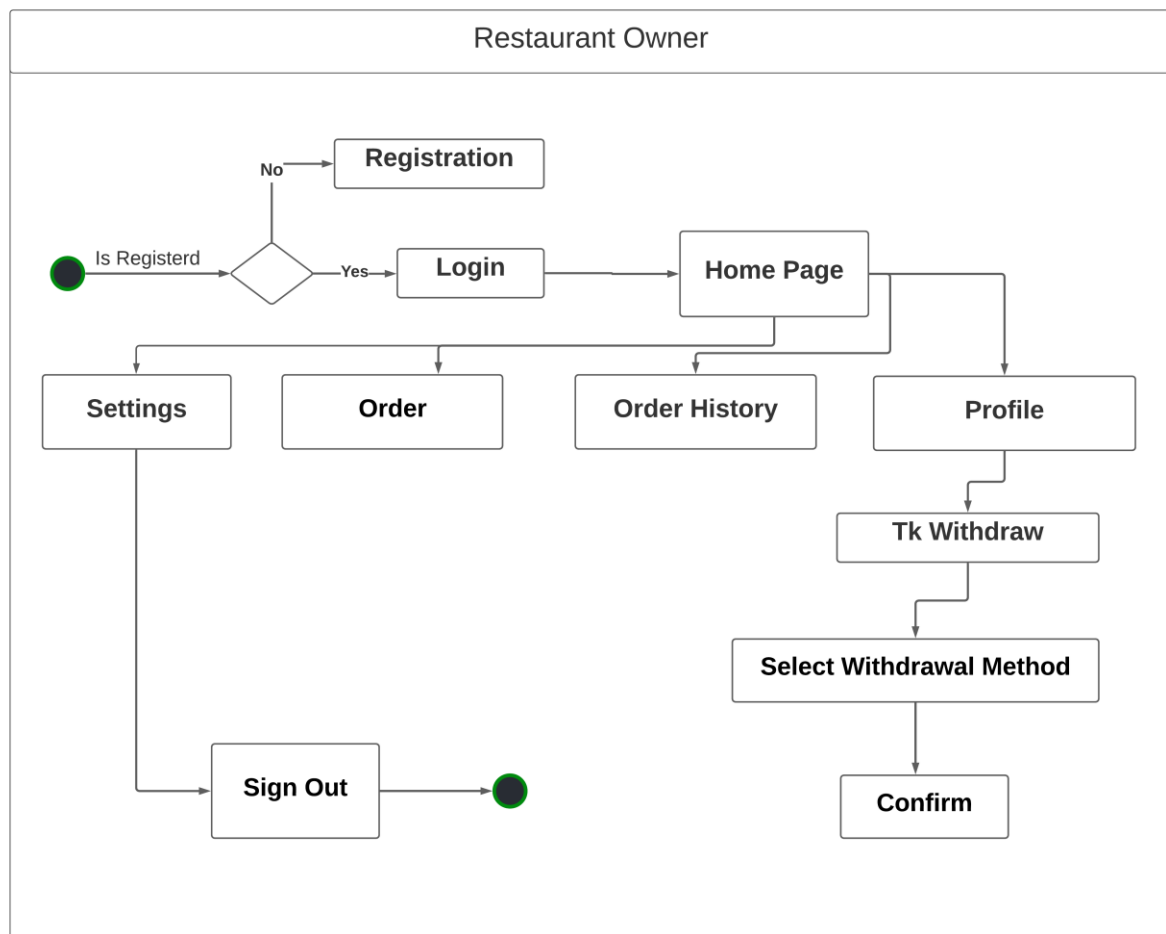


Figure 7: Restaurant Owner Activity Diagram of Foodpatrons

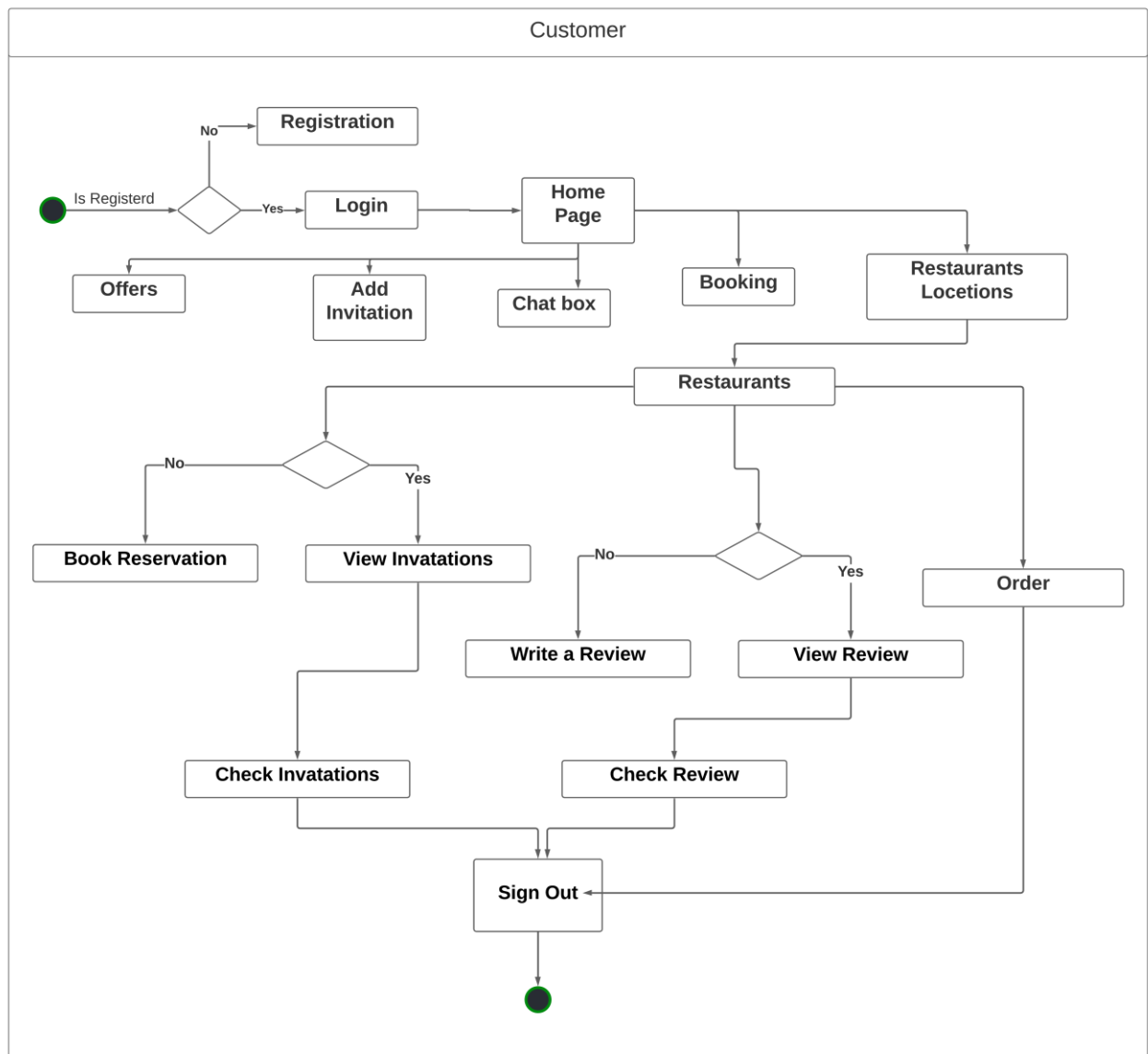


Figure 8: Customer Activity Diagram of Foodpatrons

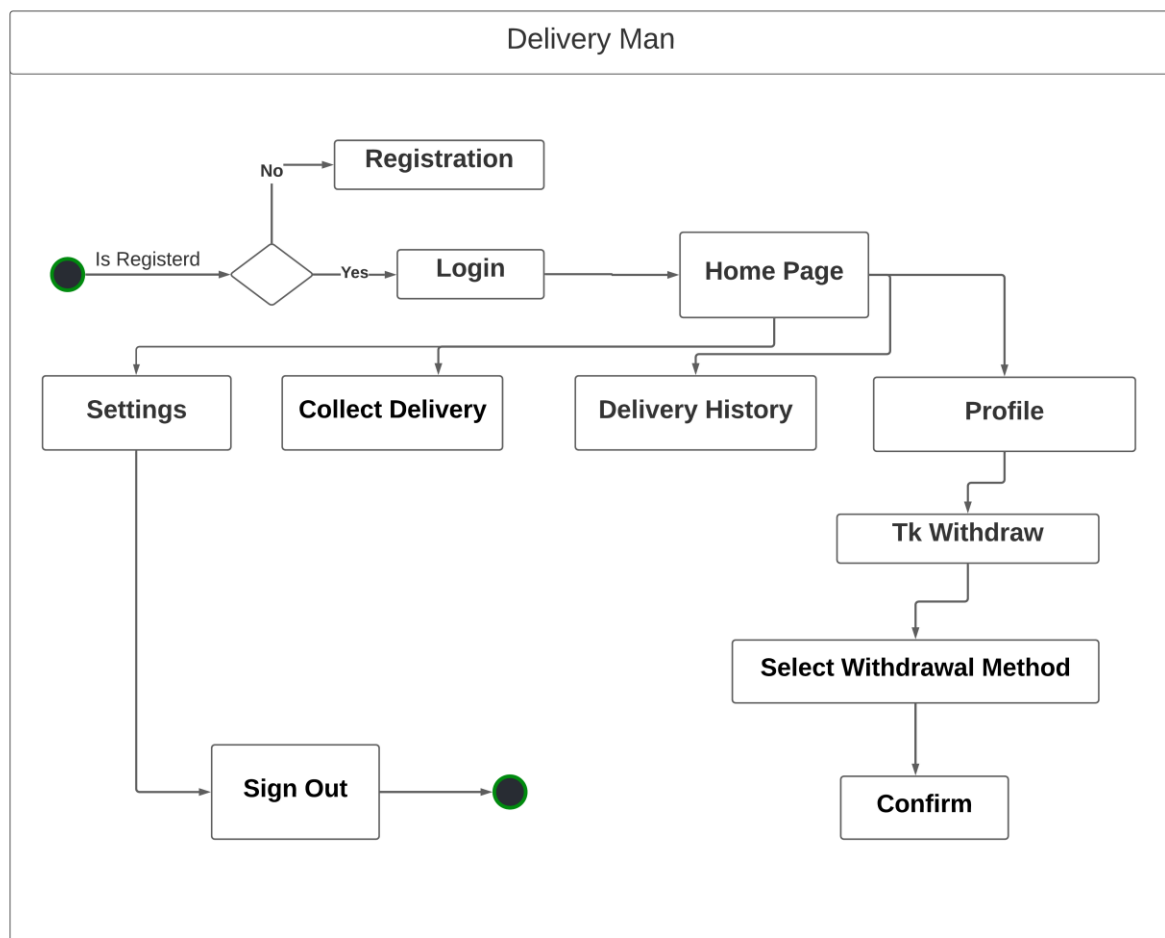


Figure 9: Delivery Man Activity Diagram of Foodpatrons

8.5 Requirement Catalogue

A requirement catalogue is a list that contains all of a project's identified needs. I'm going to enter the identified needs into the requirement catalog in the normal manner.

Customer app Requirement Catalogue

Functional Requirement

- Create an account
- Manage their account
- Select an item from the menu.
- Customize options for a selected item.
- Customer can see Restaurants those are close to them
- Customer can choose their partner as they want to
- Customer can book Restaurants according to their date
- Customer can chat with their partner through this app

- Customer can send invitation to their friends to go on a date
- Customer can see restaurants offers present time
- Customer can create invitation
- Customer can review a restaurants
- Customer can search restaurants
- Customer can place an order

Non-functional requirement

- The system should be able to run the program on a variety of platforms, including different versions of the operating system and mobile devices.
- Usability (Efficiency of use, Intuitiveness, Low perceived workload)
- The system must be able to do authentication and payment through bankcard.
- The user will be able to pay using a secure bank payment method.
- The system must be capable of encrypting and decrypting data for the user's login password.

Description	Target Value	Acceptance Value	Comment
Login/Registration per day	1000 per day.	8500	

Table 1: Customer app Requirement Catalogue of Foodpatrons

Delivery man app Requirement Catalogue

Functional Requirement

- Create an account
- Manage their account
- The delivery lad will be allowed to begin his duty.
- The delivery boy will be able to choose between Online/Offline/Break modes.
- The delivery boy will be able to obtain feedback from the consumer.
- The delivery boy can withdraw money in Bikash, Nogod, Visa card, Mastercard

Non-functional requirement

- The system should be able to run the program on a variety of platforms, including different versions of the operating system and mobile devices.
- Usability (Efficiency of use, Intuitiveness, Low perceived workload)
- The system must be able to do authentication and payment through bankcard.
- The user will be able to pay using a secure bank payment method.
- The system must be capable of encrypting and decrypting data for the user's login password.

Description	Target Value	Acceptance Value	Comment
Login/Registration per day	1000 per day.	8500	

Table 2: Delivery man app Requirement Catalogue of Foodpatrons

Restaurants Owner app Requirement Catalogue

Functional Requirement

- Create an account
- Manage their account
- Restaurants Owner will manage food orders.
- Restaurants Owner will manage book reservation.
- Restaurants Owner will manage food orders.
- Restaurants Owner can withdraw money in Bikash, Nogod, Visa card, Mastercard

Non-functional requirement

- The system should be able to run the program on a variety of platforms, including different versions of the operating system and mobile devices.
- Usability (Efficiency of use, Intuitiveness, Low perceived workload)
- The system must be able to do authentication and payment through bankcard.
- The user will be able to pay using a secure bank payment method.

- The system must be capable of encrypting and decrypting data for the user's login password.

Description	Target Value	Acceptance Value	Comment
Login/Registration per day	1000 per day.	8500	

Table 3: Restaurants Owner app Requirement Catalogue of Foodpatrons

8.6 Prioritized Requirement List (PRL)

To create a priority list of recognized requirements, I used the MoSCoW prioritizing approach. The Foodpatrons System's priority requirement list is as follows:

Must Have Requirements

SL	Requirement	Priority
1.	Registration and sign in for all users	Must
2.	Restaurants booking system	Must
3.	Partner Selection System	Must
4.	Food order and delivery system	Must
5.	Sent Invitation system	Must
6.	Location tracking and notification system	Must

SL	Requirement	Priority
----	-------------	----------

1.	Anonymous Order	Should
2.	Payment system	Should
3.	Money withdraw system	Should

Should Have Requirements

Could Have Requirements

SL	Requirement	Priority
1.	Restaurant search system	Could
2.	Show restaurant offers	Could
3.	Review system	Could

8.7 Prototype of new system

Foodpatrons prototype system is given bellow:

Customer app

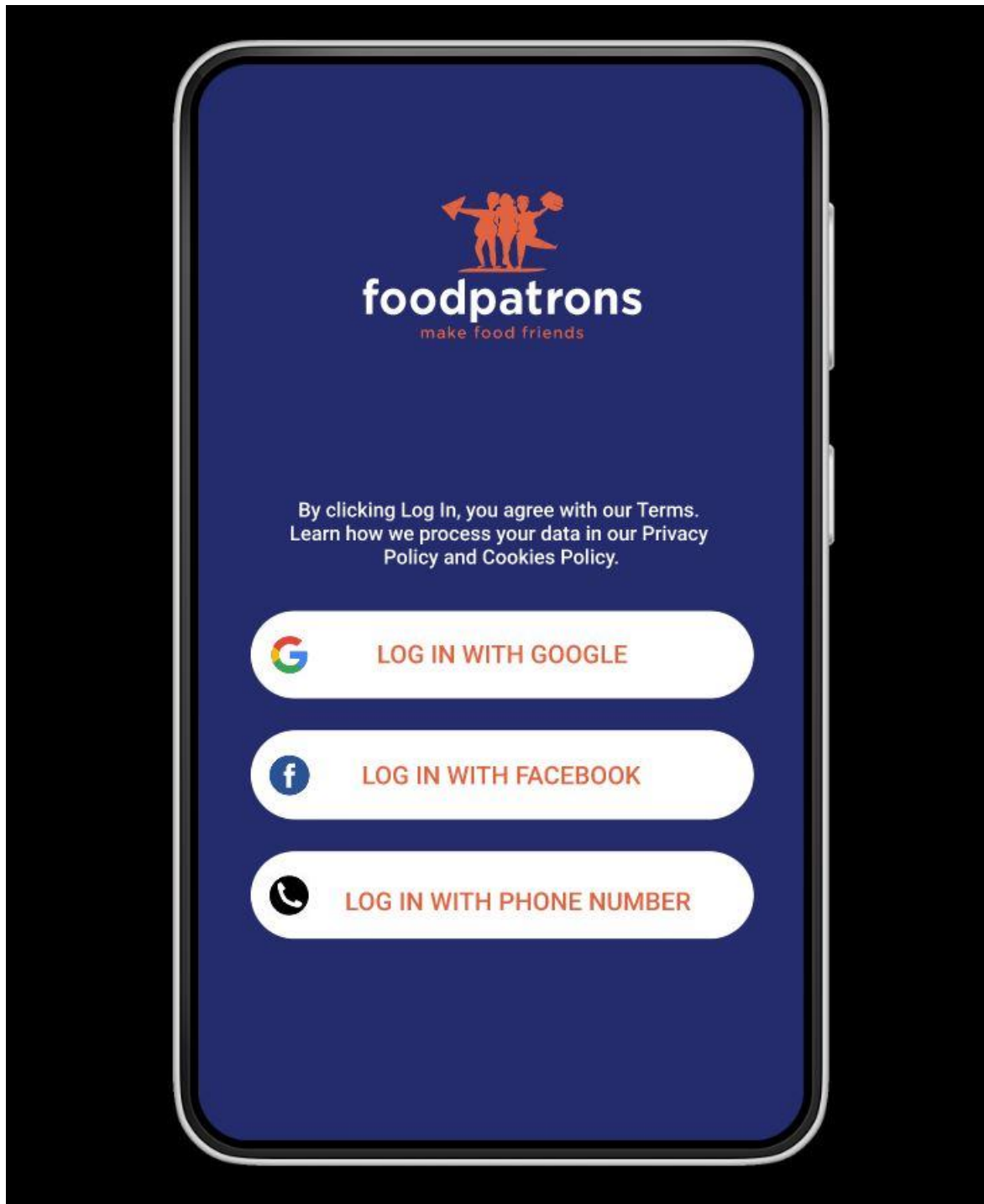


Figure 10: Customer app Login option page of Foodpatrons

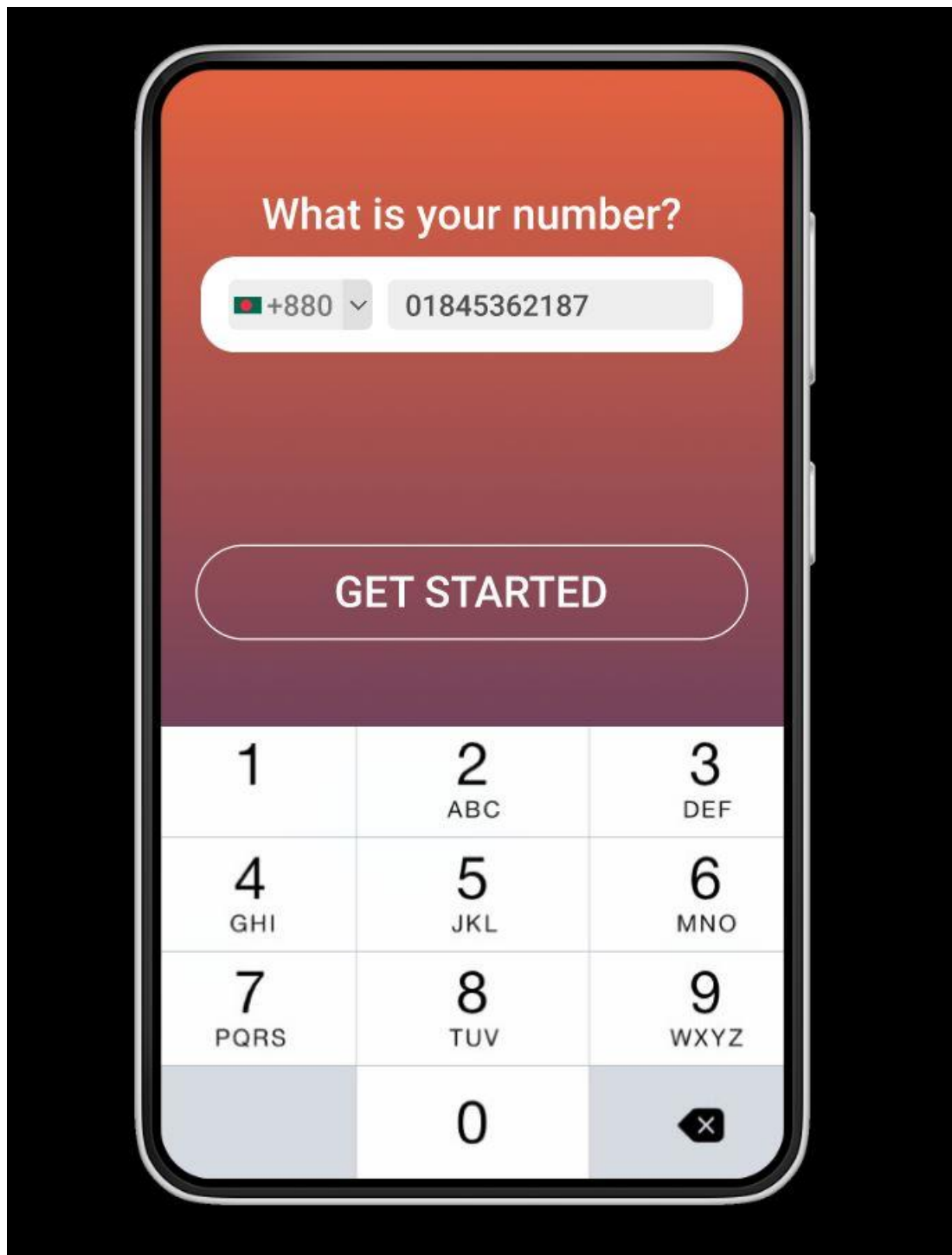


Figure 11: Customer app login with phone number page of Foodpatrons

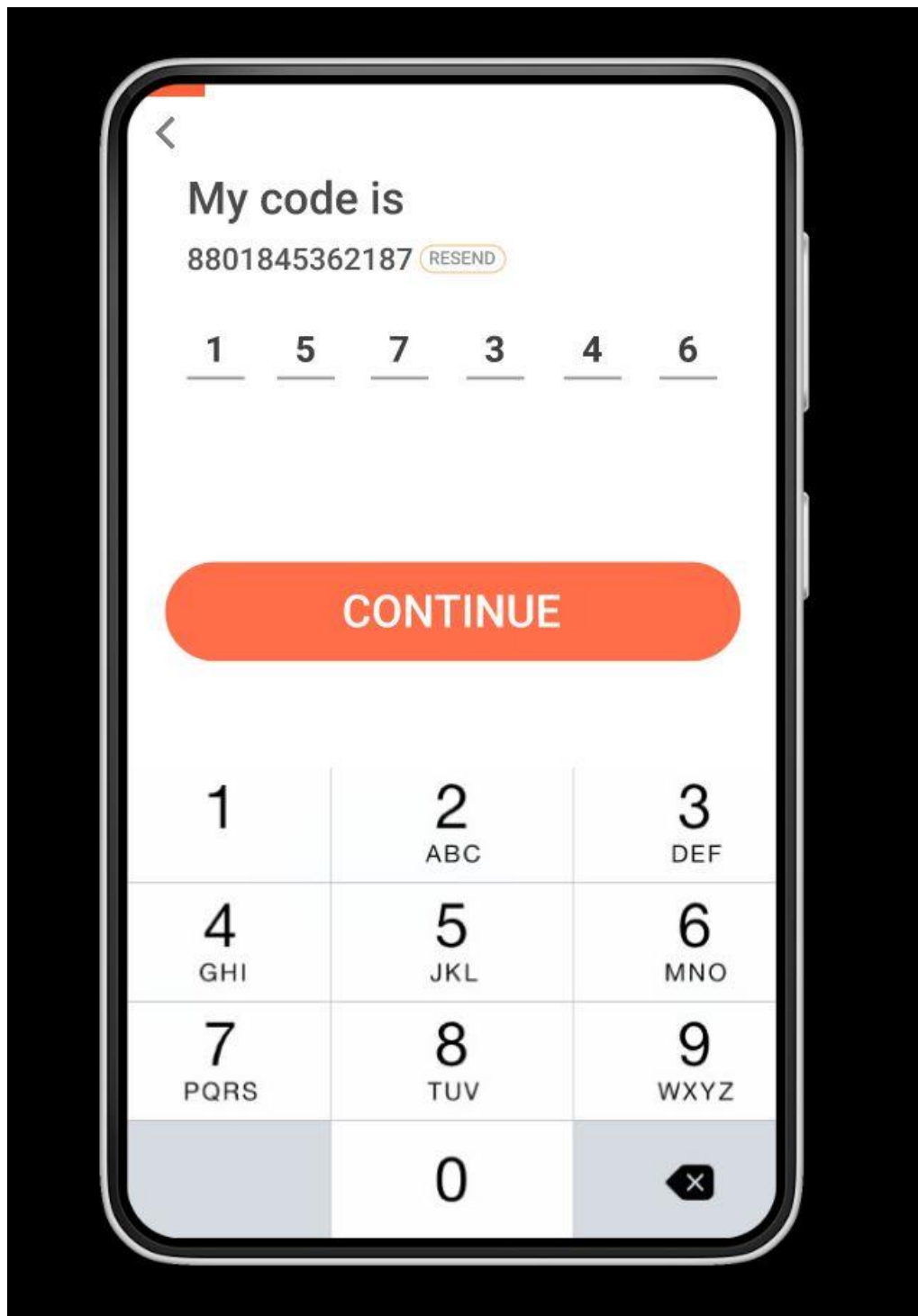


Figure 12: Customer app OTP page of Foodpatrons

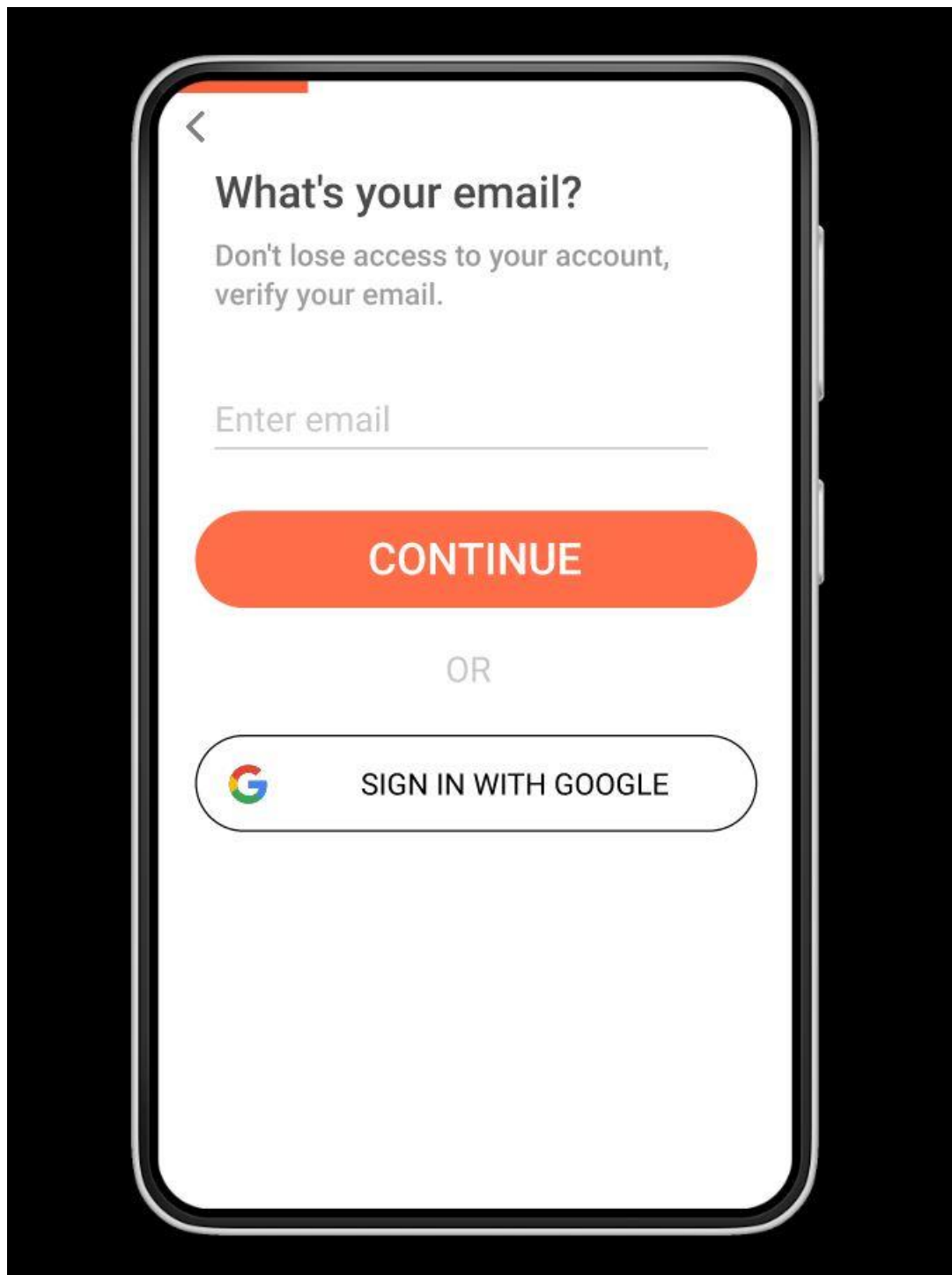


Figure 13: Customer app Email entry page of Foodpatrons

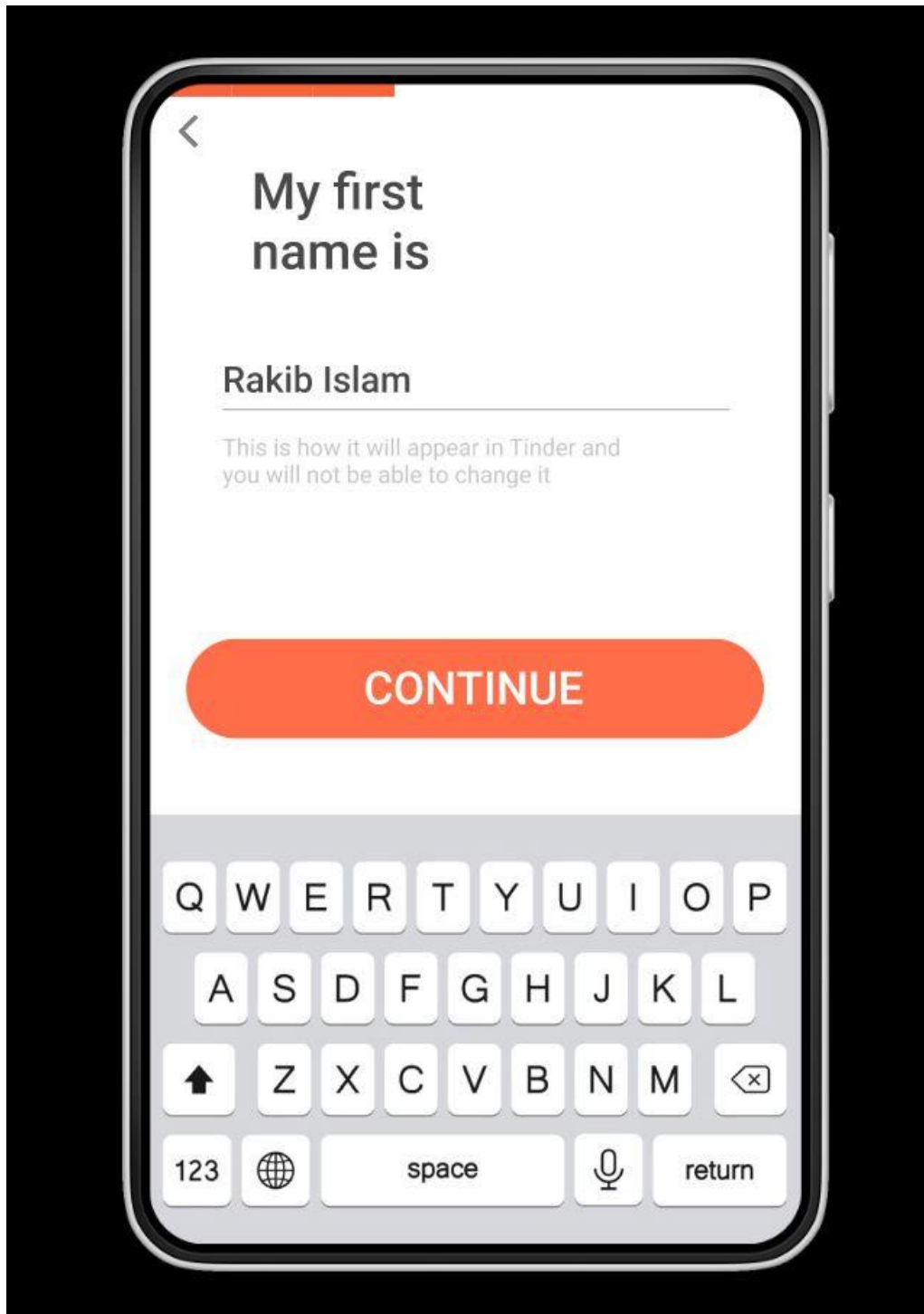


Figure 14: Customer app Name entry page of Foodpatrons

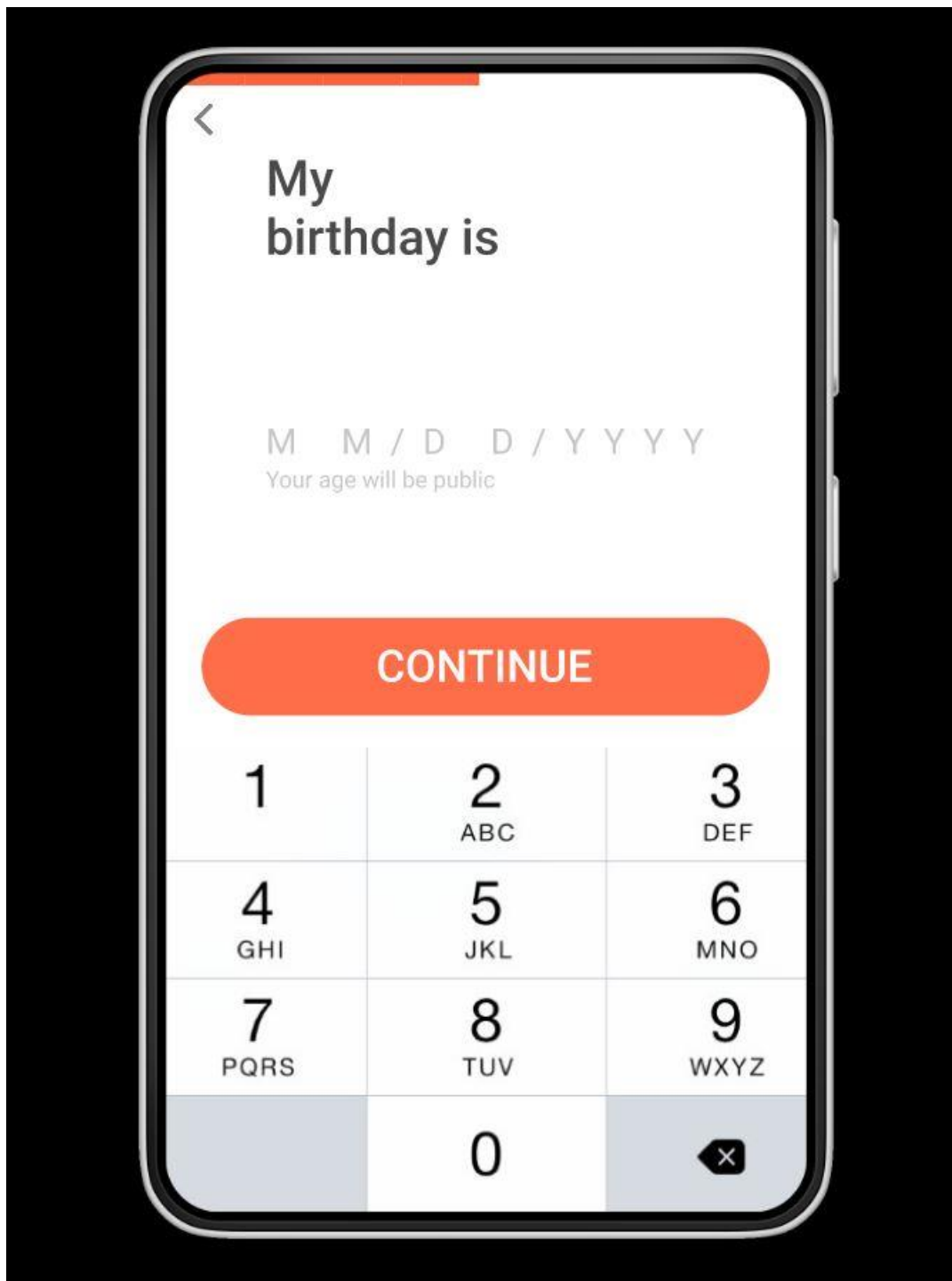


Figure 15: Customer app Birth date page of Foodpatrons

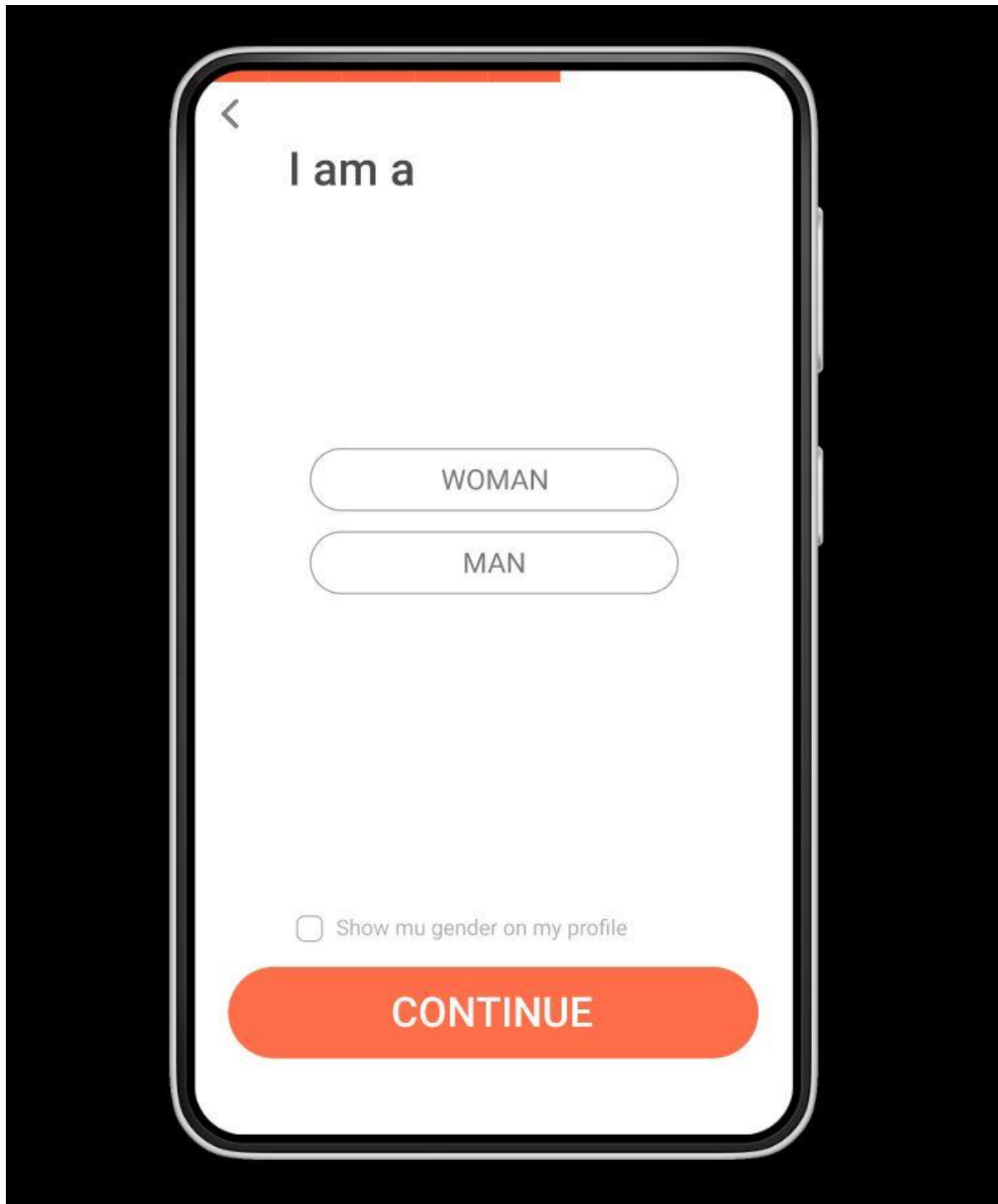


Figure 16: Customer app Gender selection page of Foodpatrons

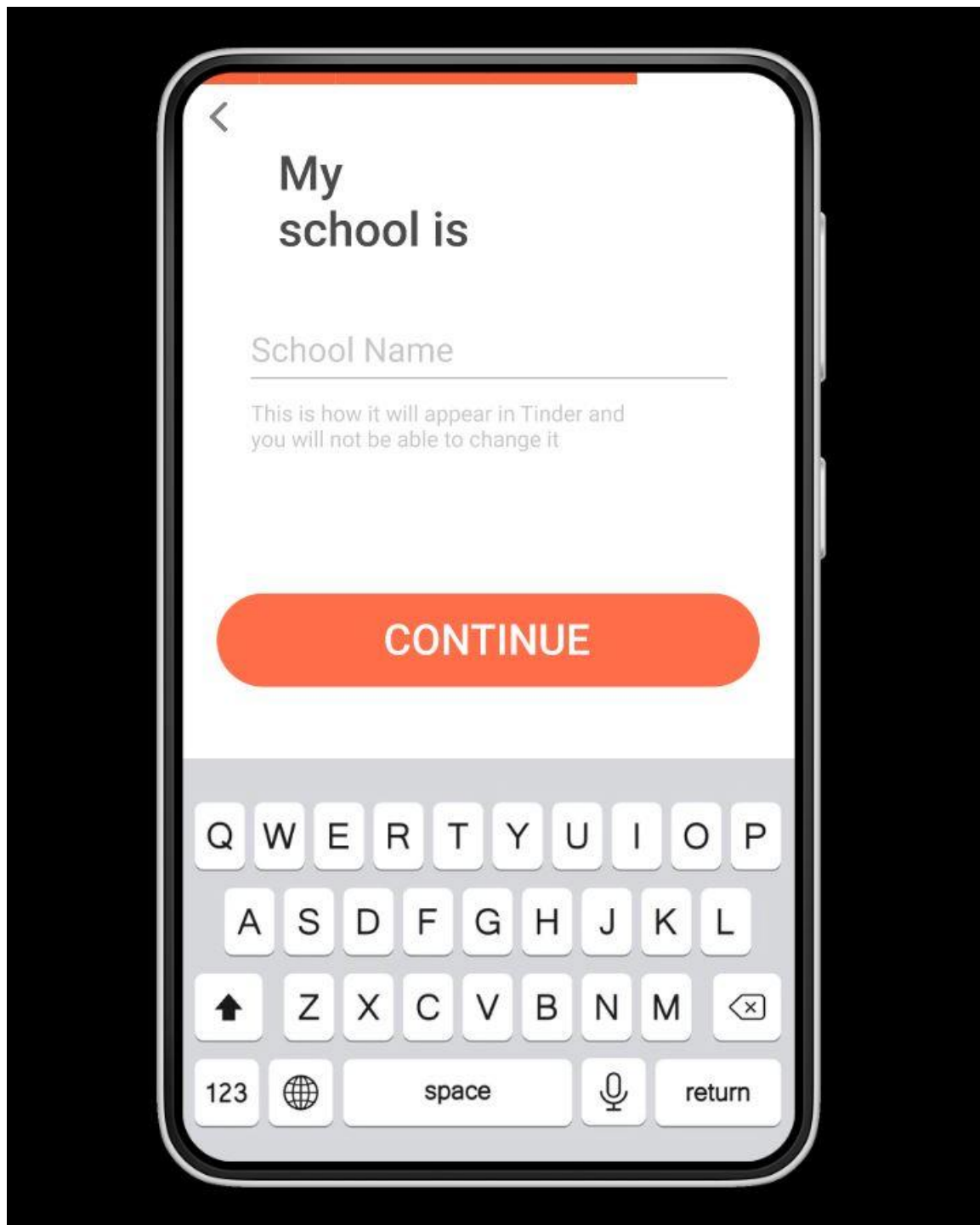


Figure 17: Customer app School name page of Foodpatrons

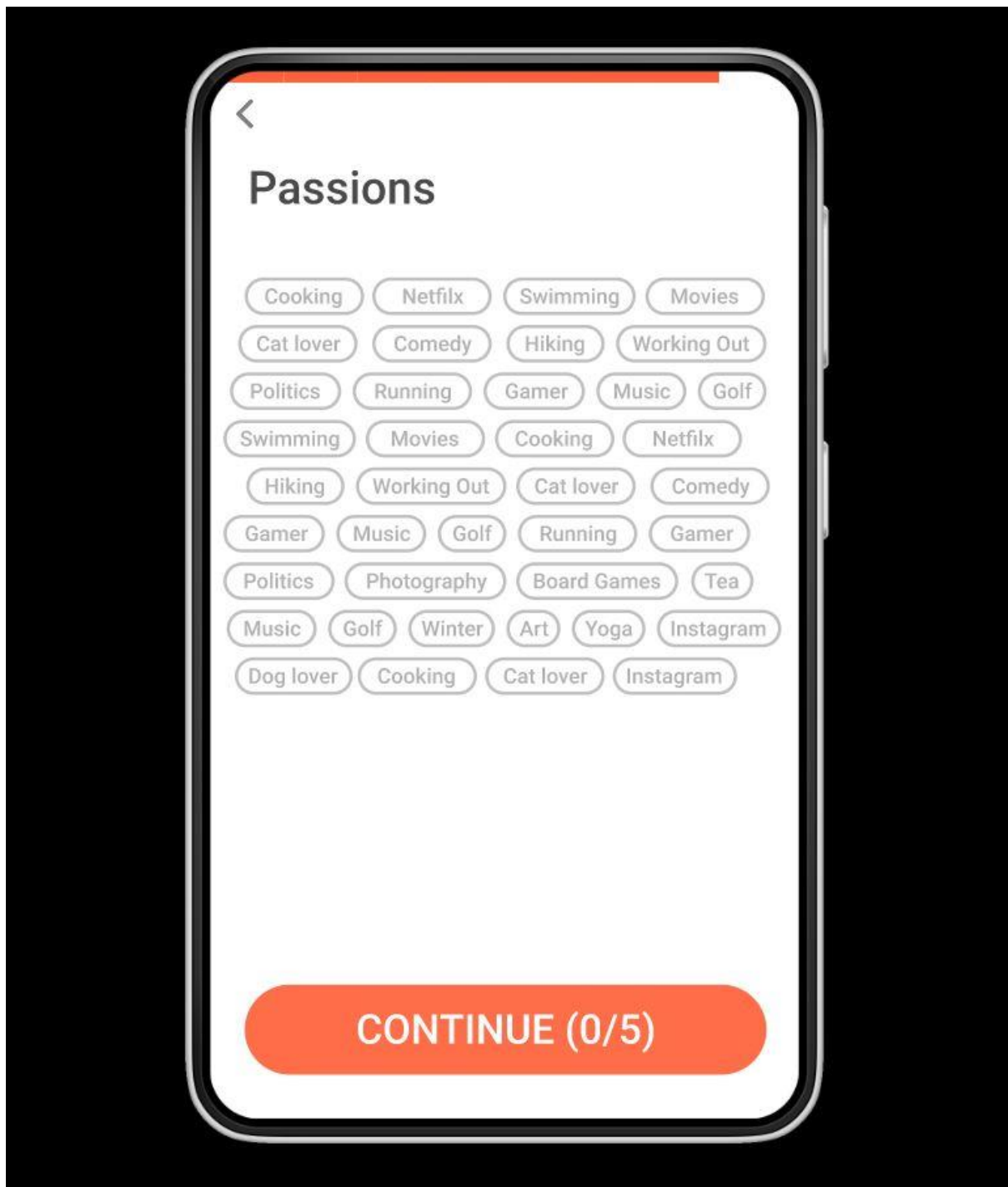


Figure 18: Customer app Passion's page of Foodpatrons

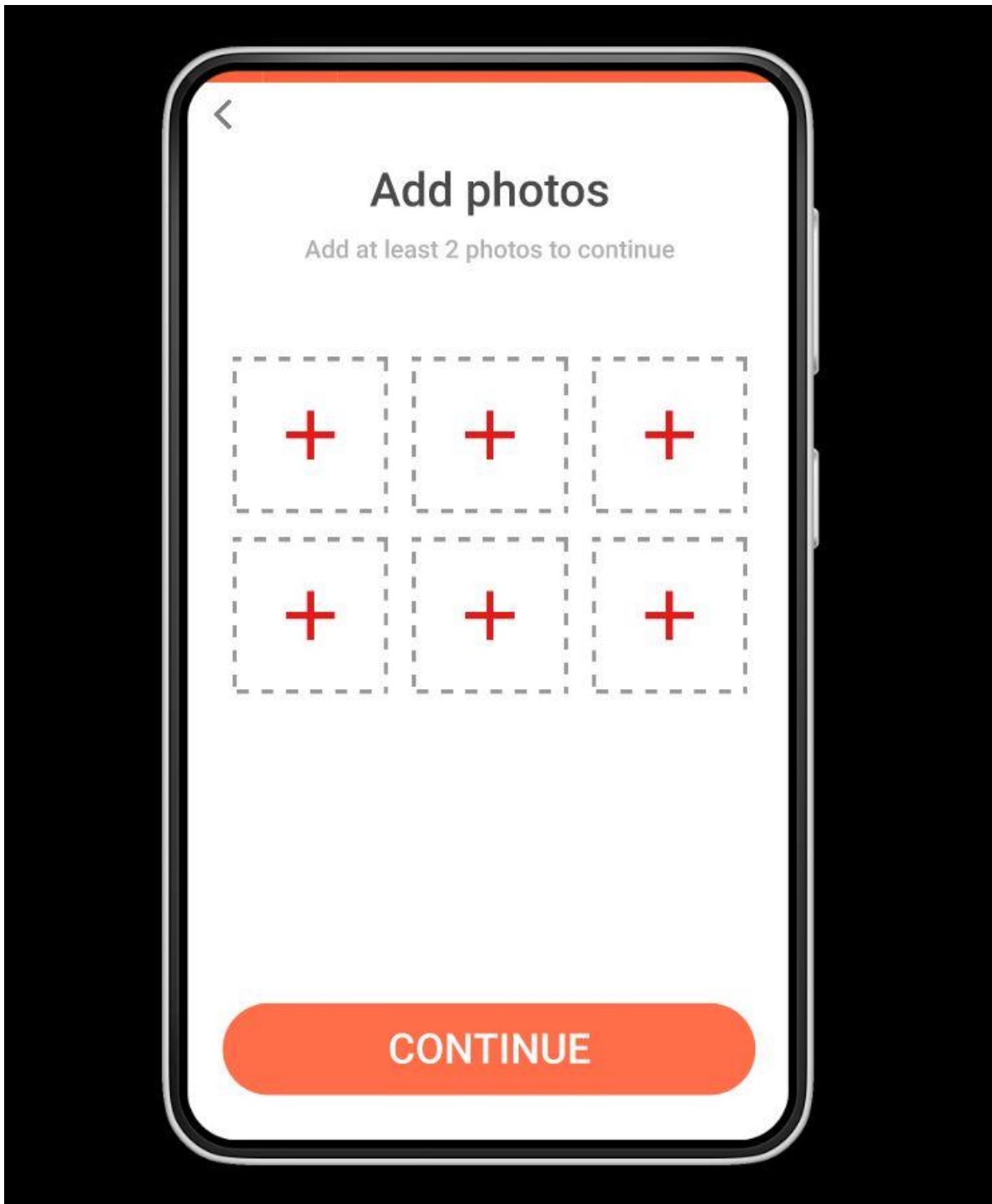


Figure 19: Customer app Add photos page of Foodpatrons

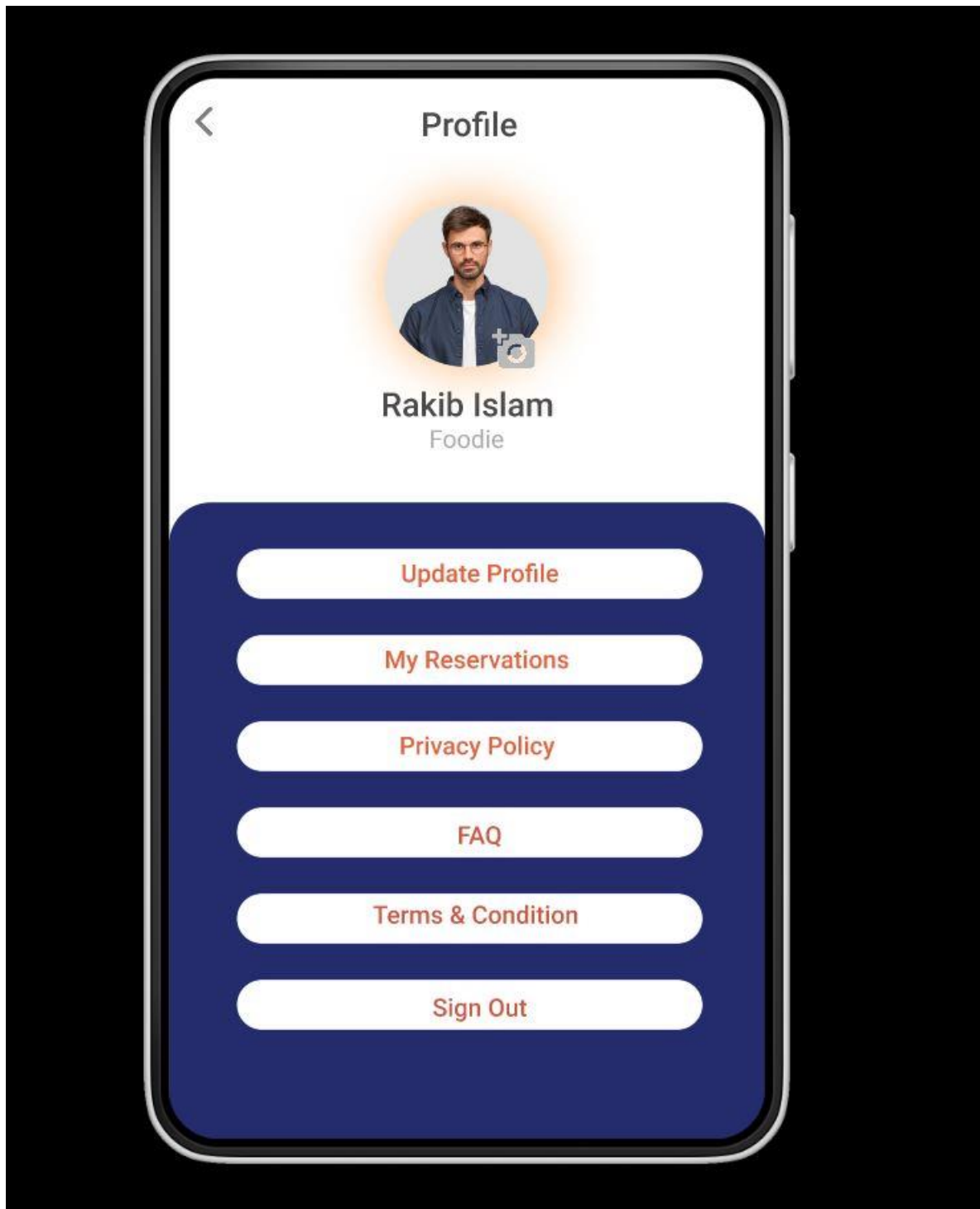


Figure 20: Customer app profile page of Foodpatrons

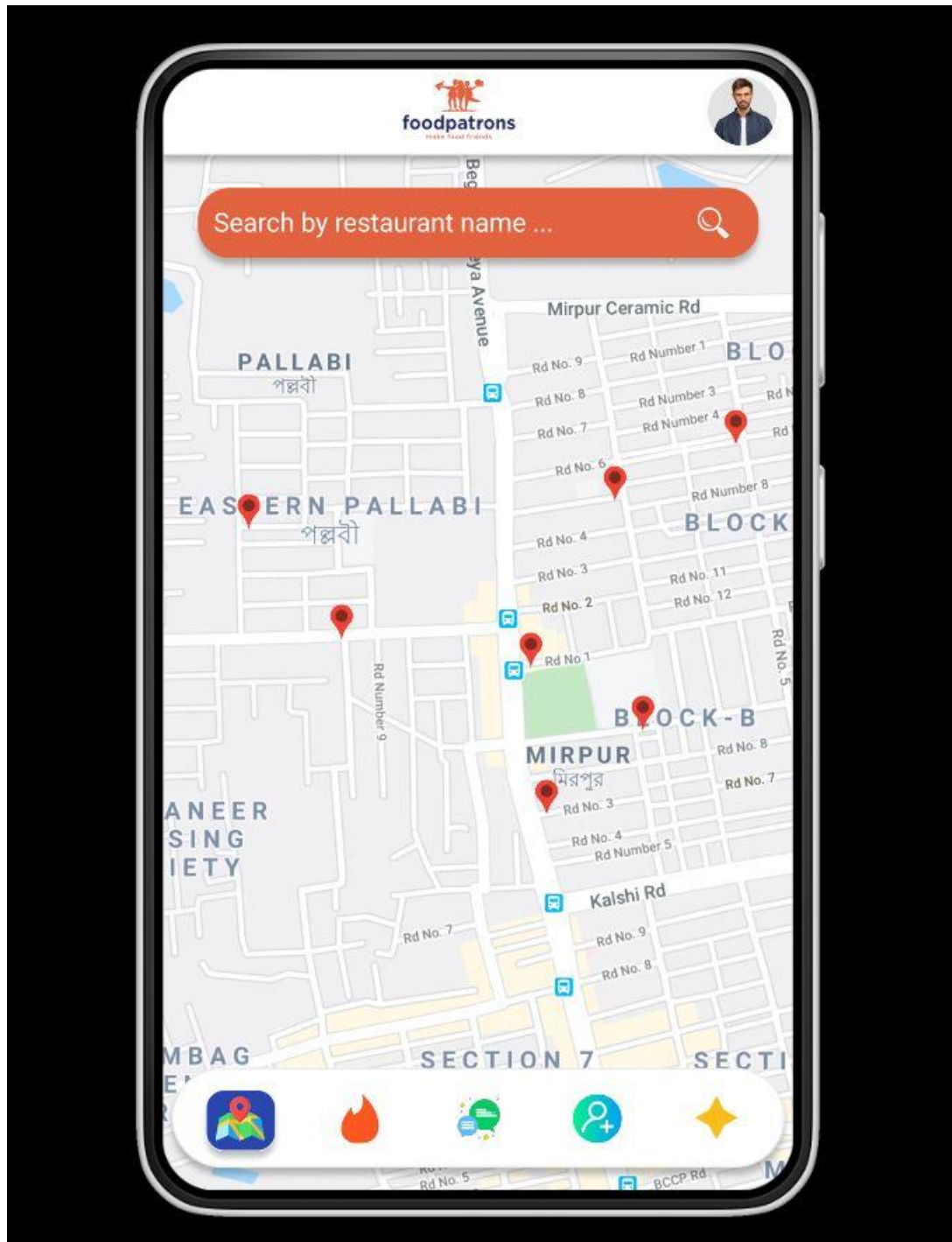


Figure 21: Customer app home page of Foodpatrons

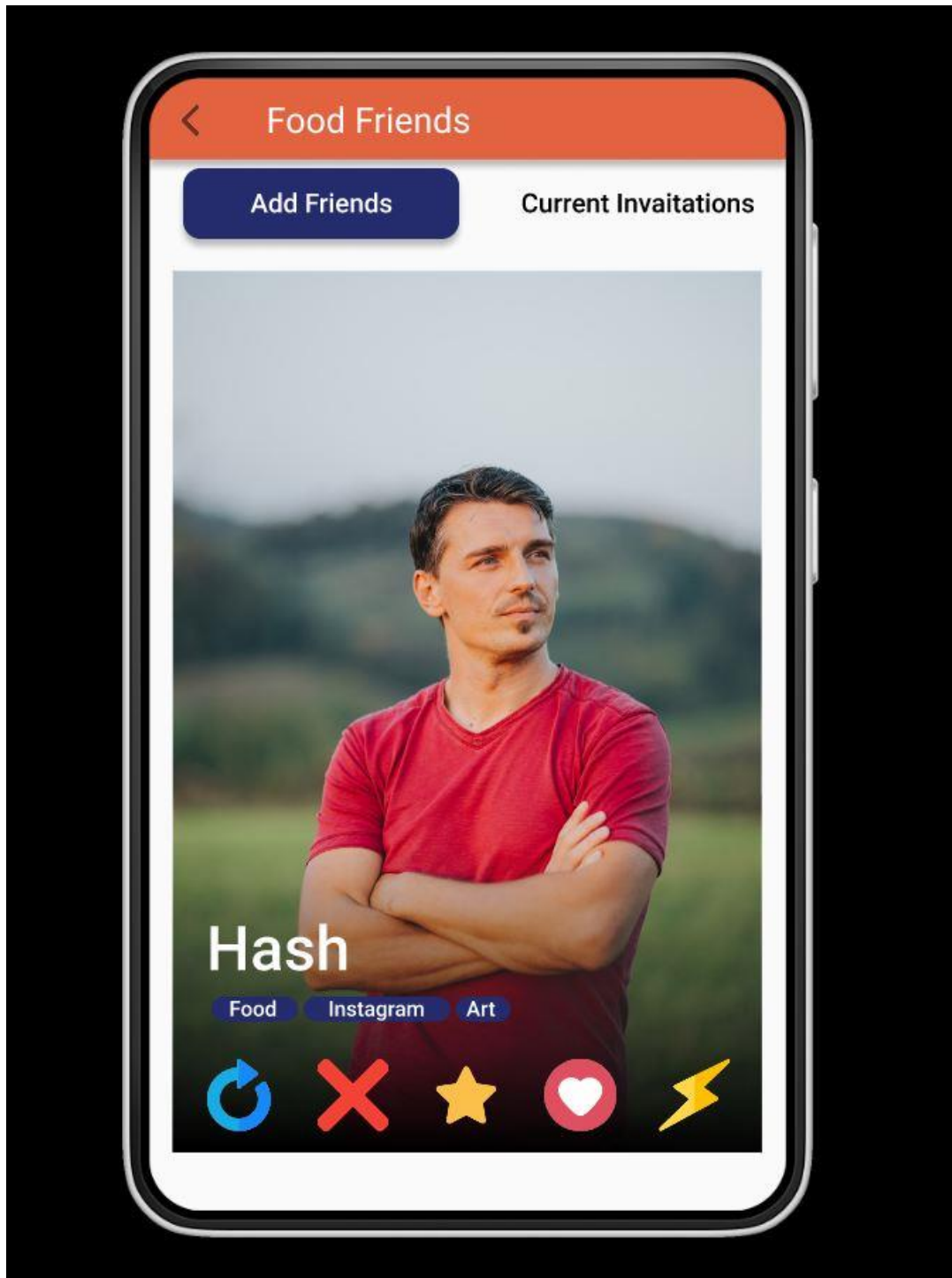


Figure 22: Customer app Partner selection page of Foodpatrons

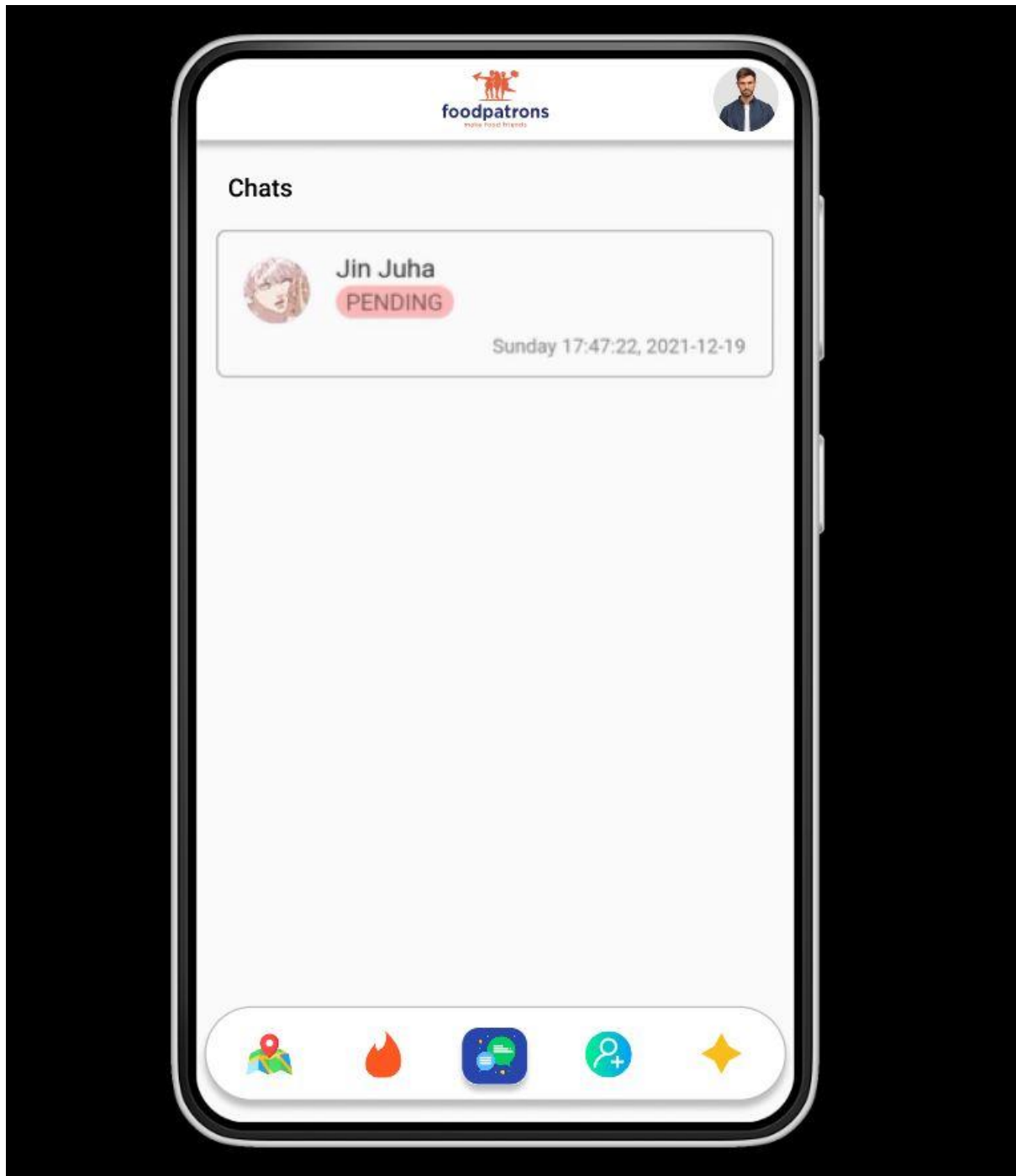


Figure 23: Customer app Chat box page of Foodpatrons

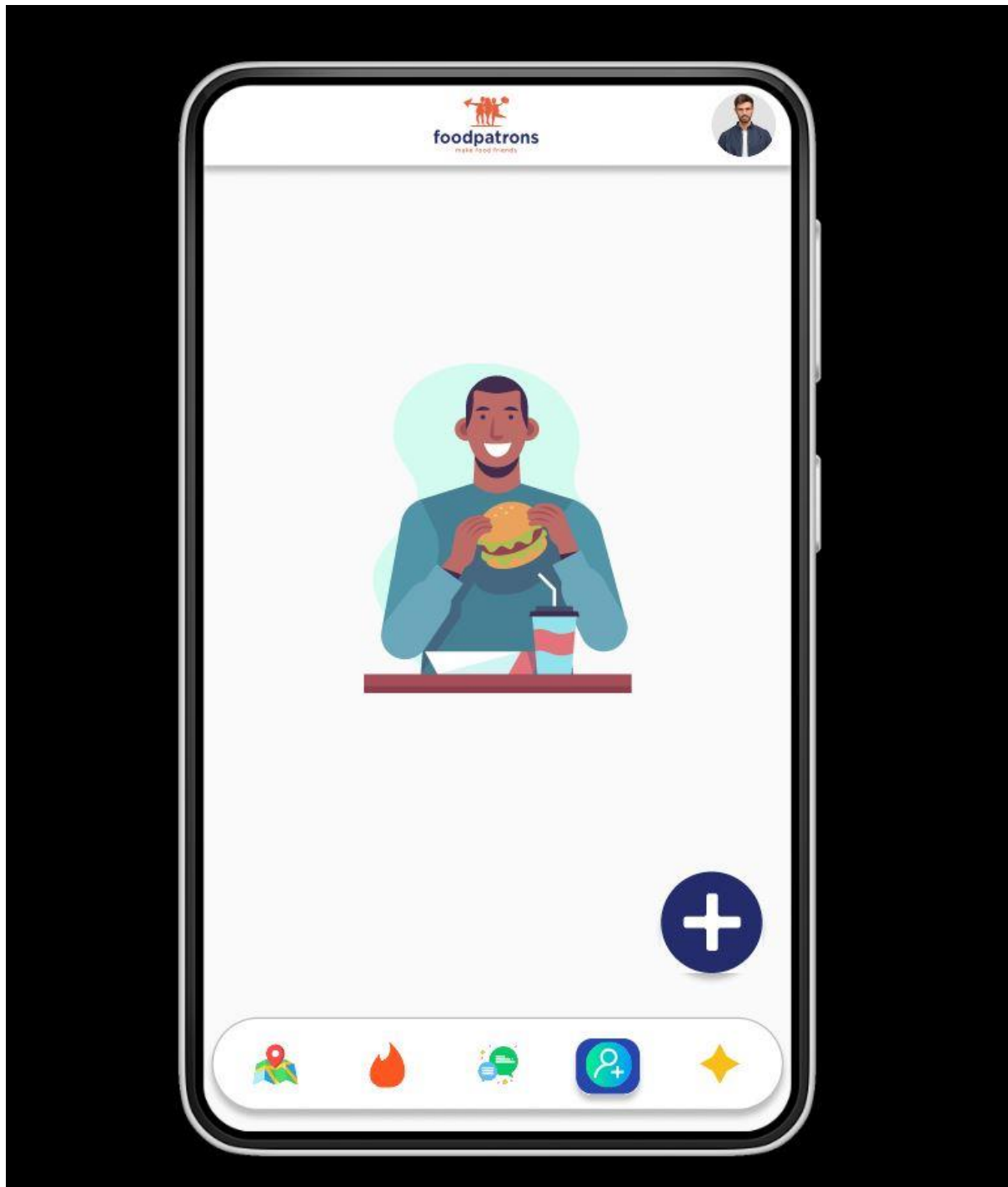
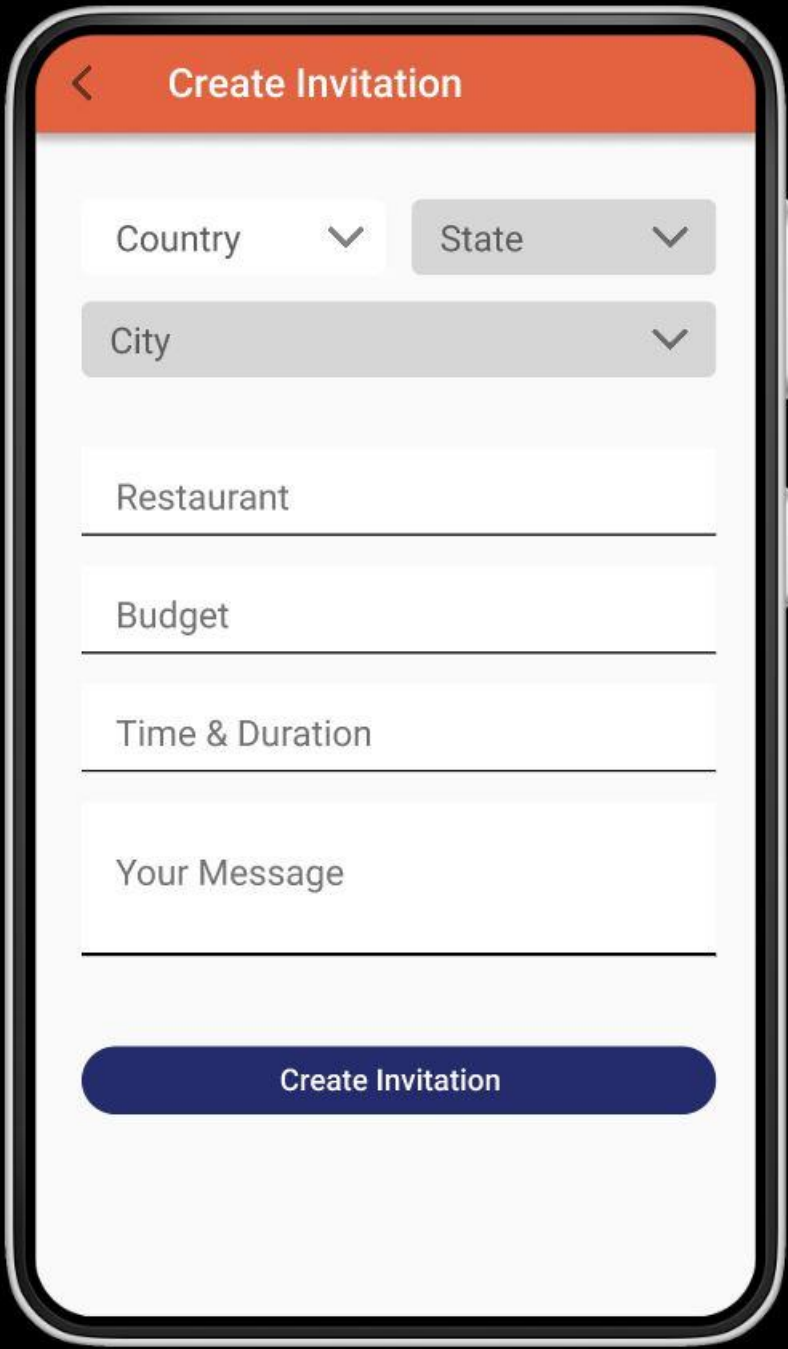


Figure 24: Customer app Add booking page of Foodpatrons



The image shows a mobile application interface for creating an invitation. The title bar at the top is orange with a back arrow and the text "Create Invitation". Below the title bar, there are three dropdown menus for location selection: "Country", "State", and "City". Each dropdown has a downward arrow icon. Below these are four text input fields: "Restaurant", "Budget", "Time & Duration", and "Your Message". Each input field has a horizontal line below it. At the bottom of the form is a dark blue button with the text "Create Invitation" in white.

Figure 25: Customer app Add booking form page of Foodpatrons

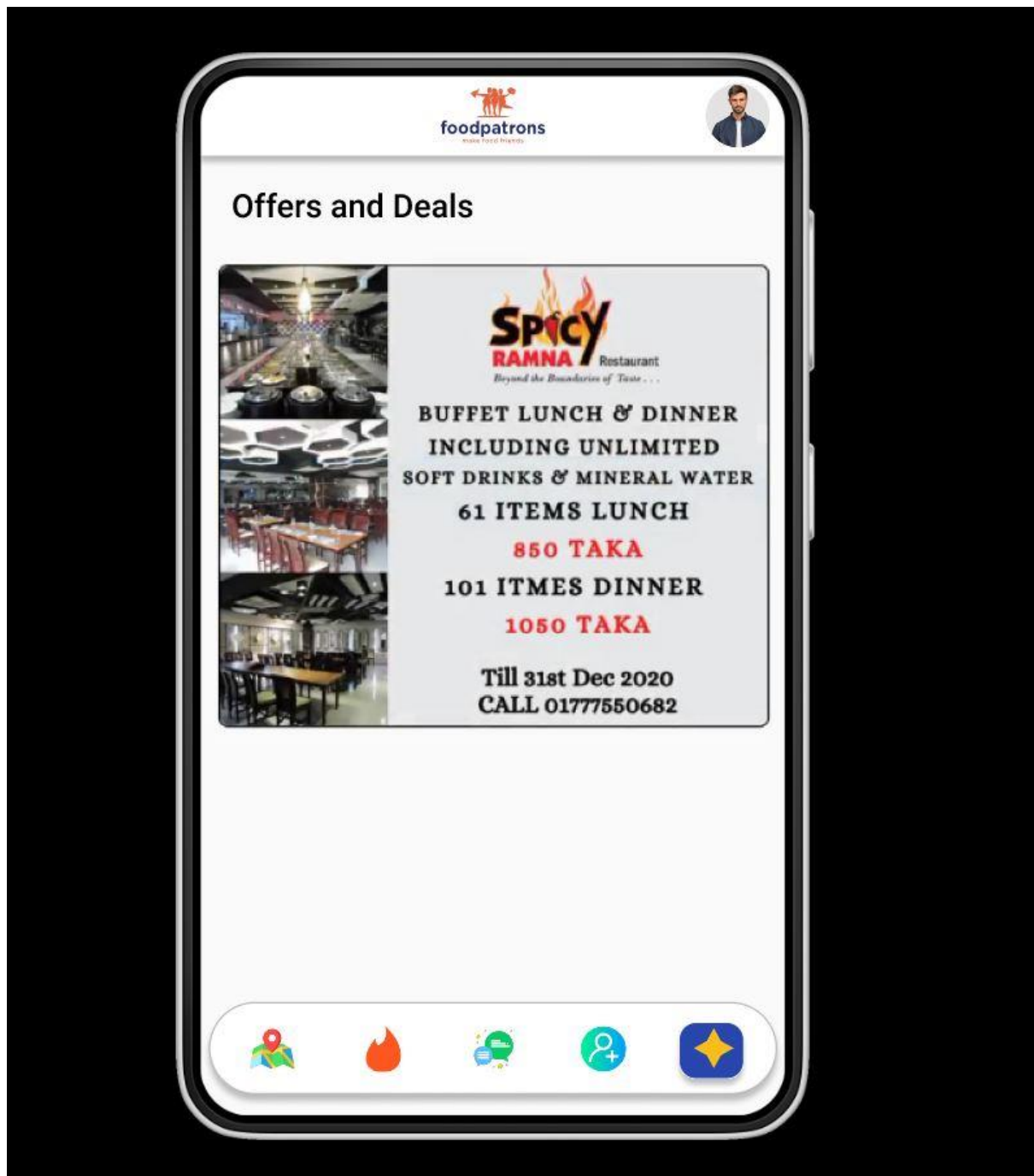


Figure 26: Customer app Restaurants offers page of Foodpatrons

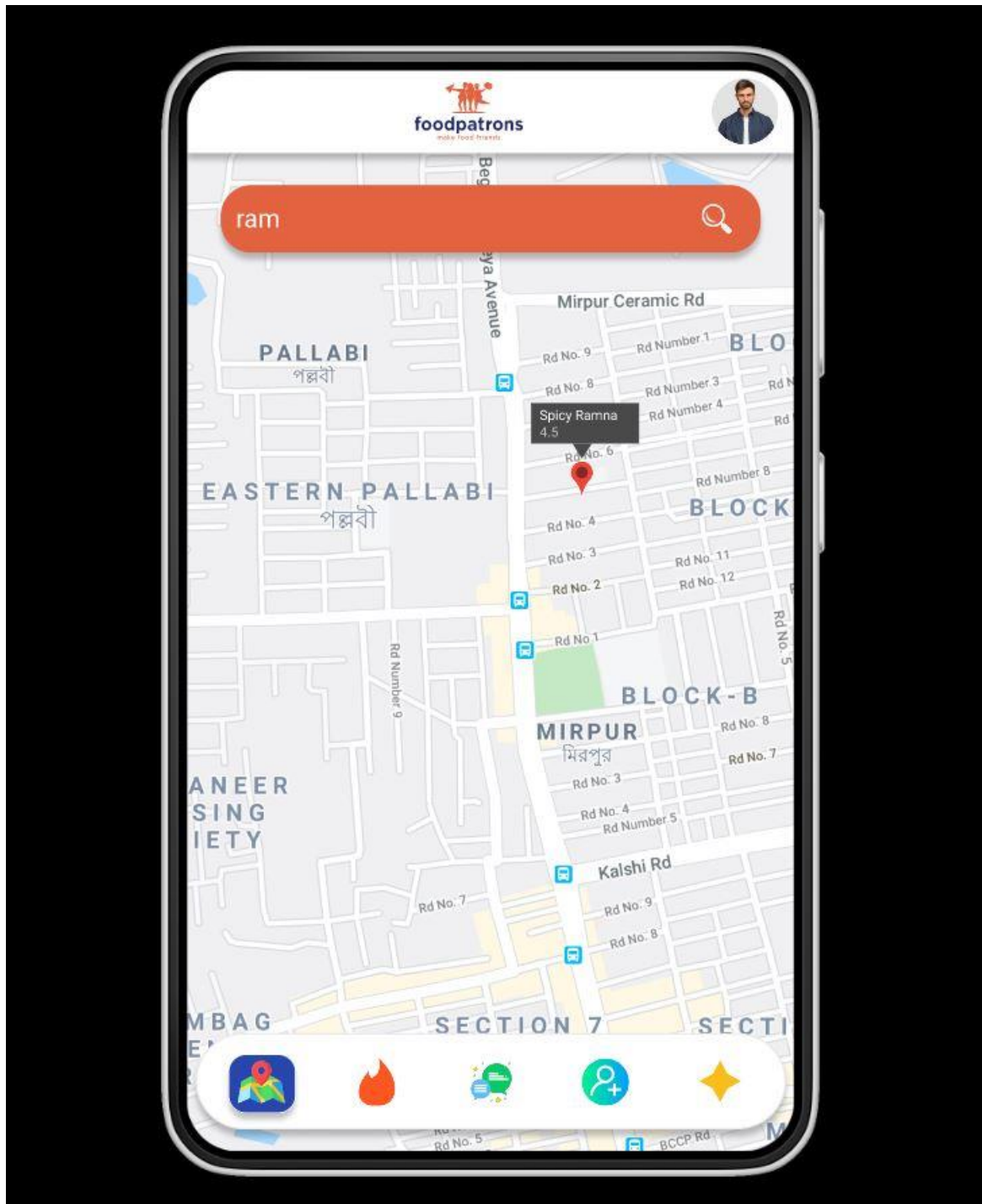


Figure 27: Customer app Restaurants Search page of Foodpatrons

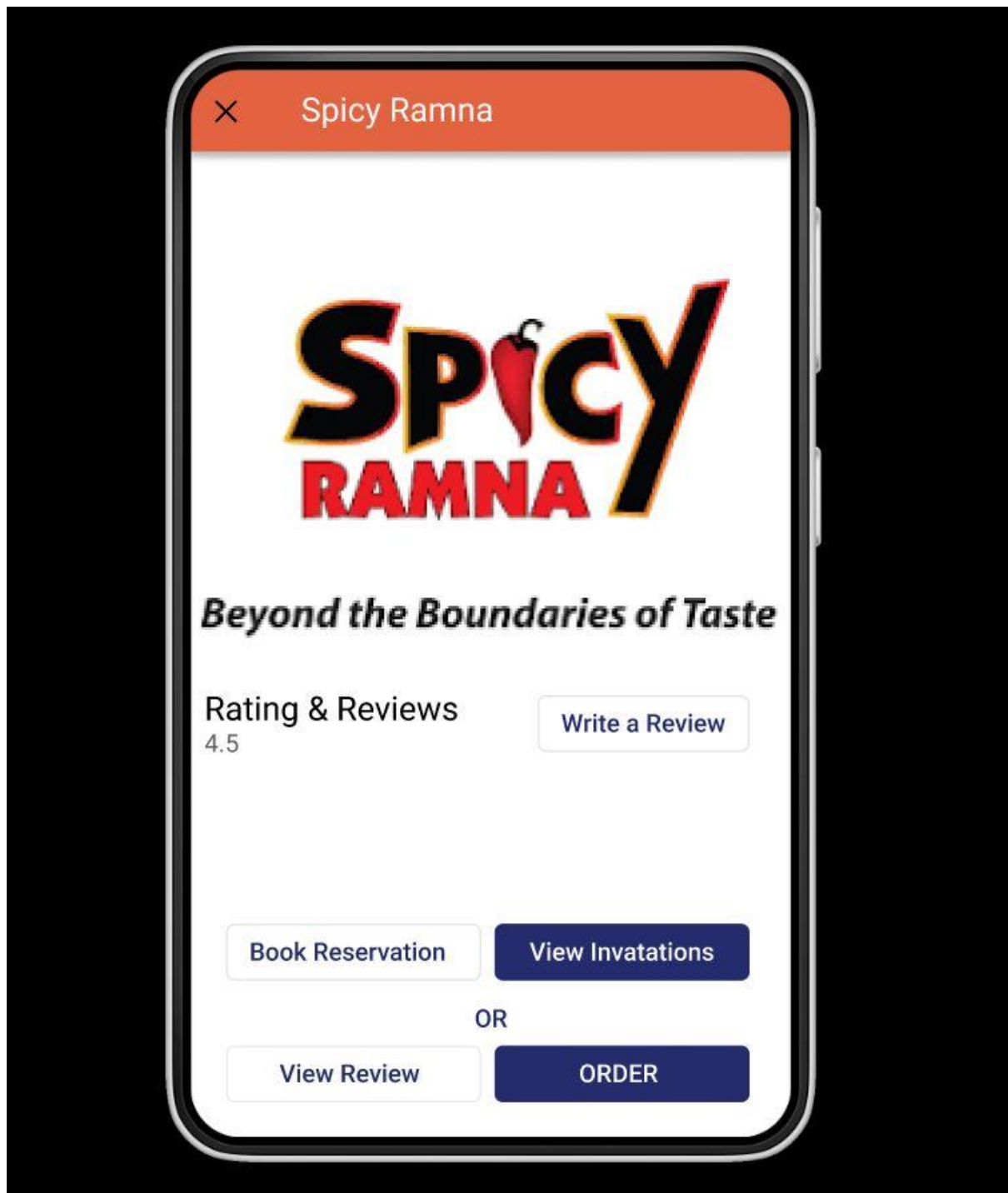


Figure 28: Customer app Restaurant's page of Foodpatrons

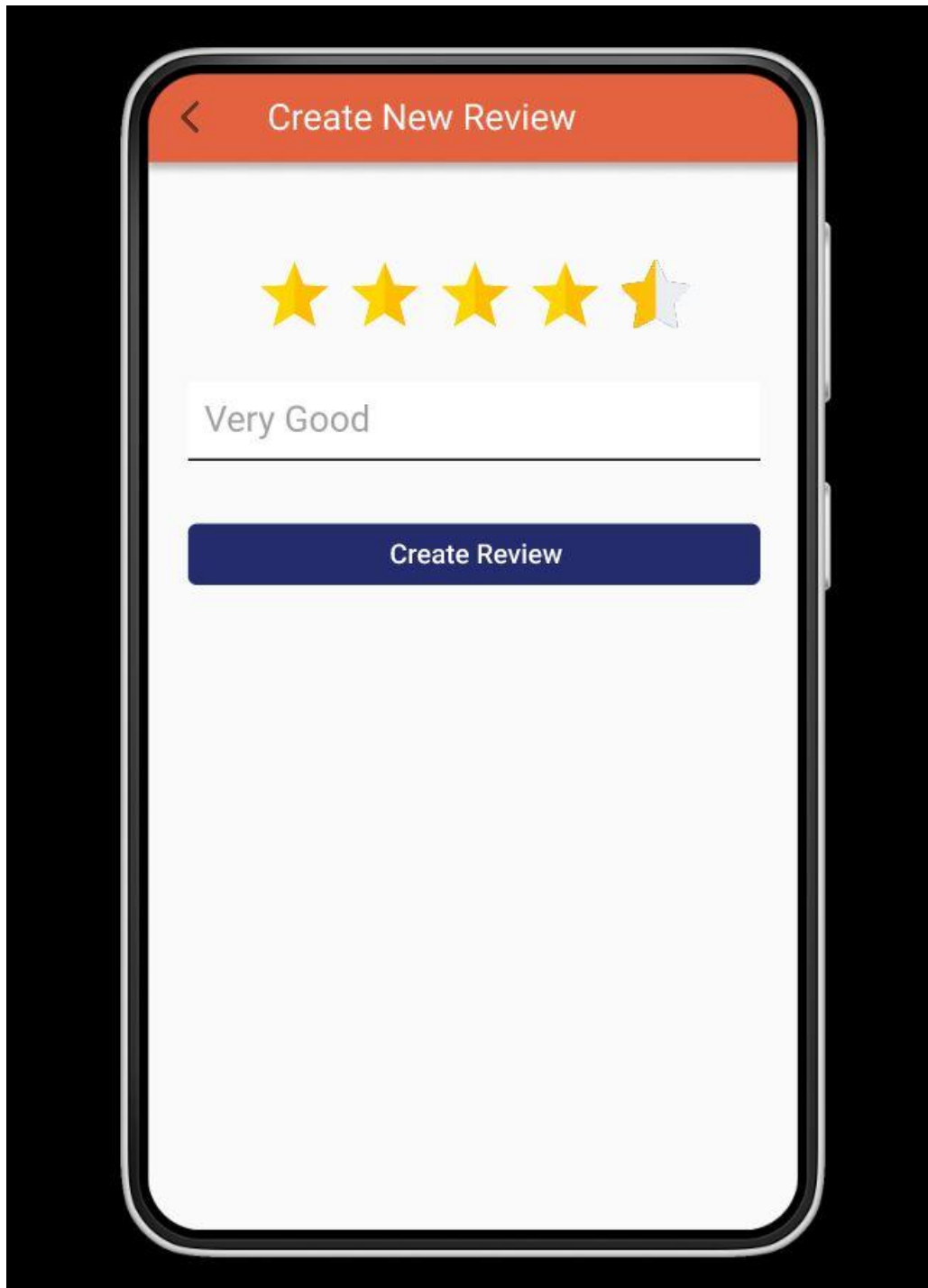


Figure 29: Customer app Restaurant's create review page of Foodpatrons

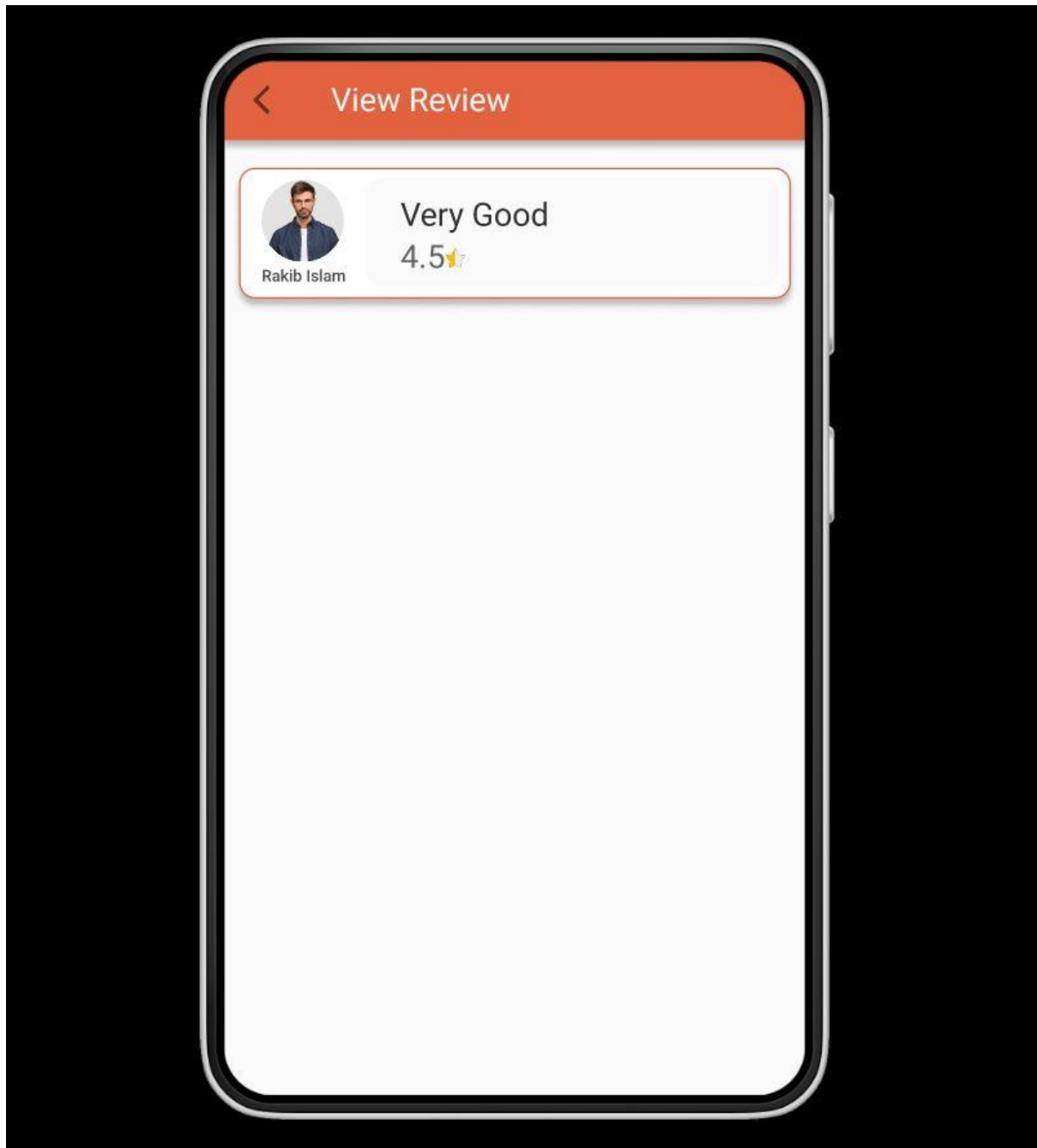


Figure 30: Customer app Restaurant's view review page of Foodpatrons

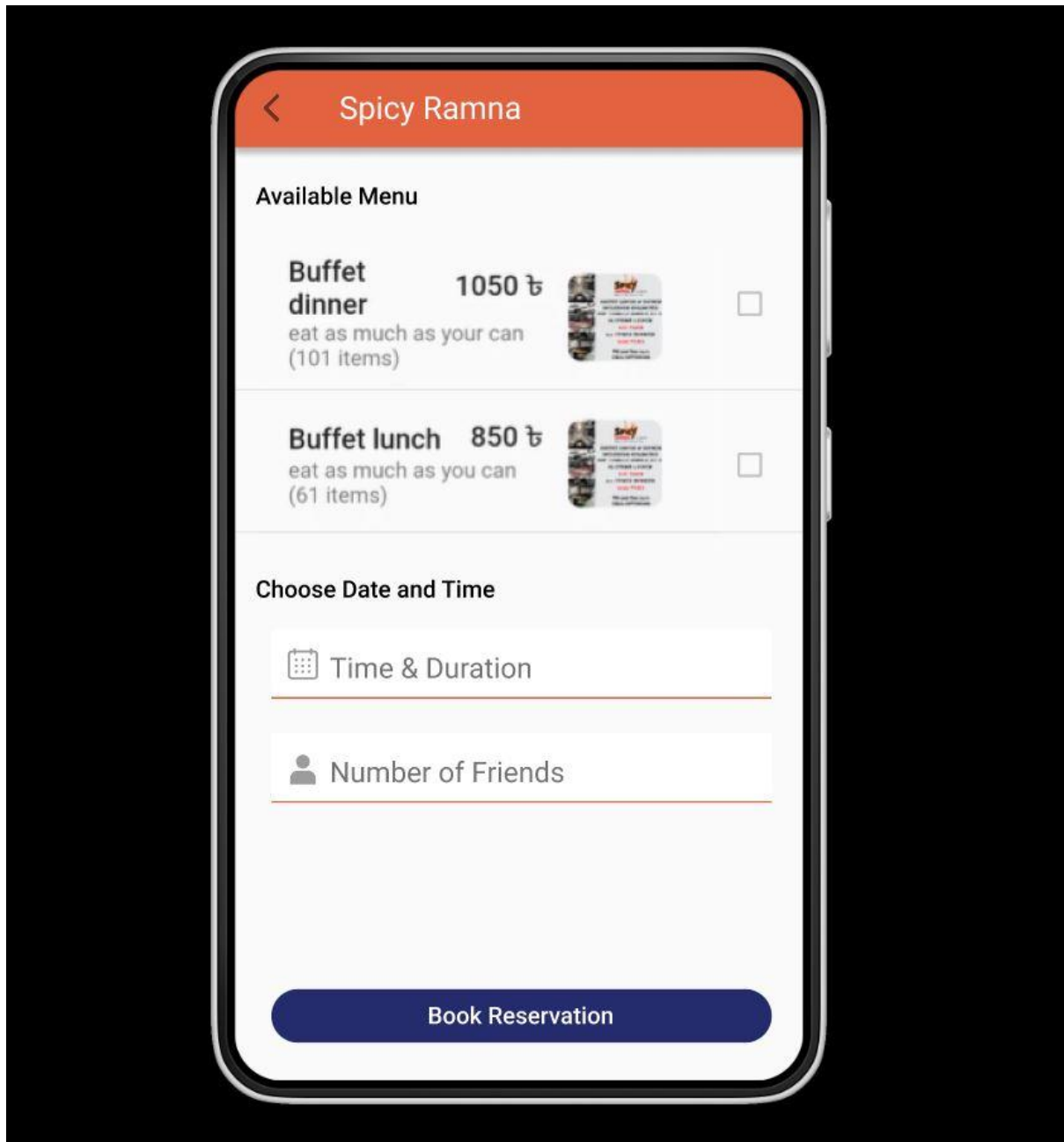


Figure 31: Customer app Restaurant's booking date form page of Foodpatrons

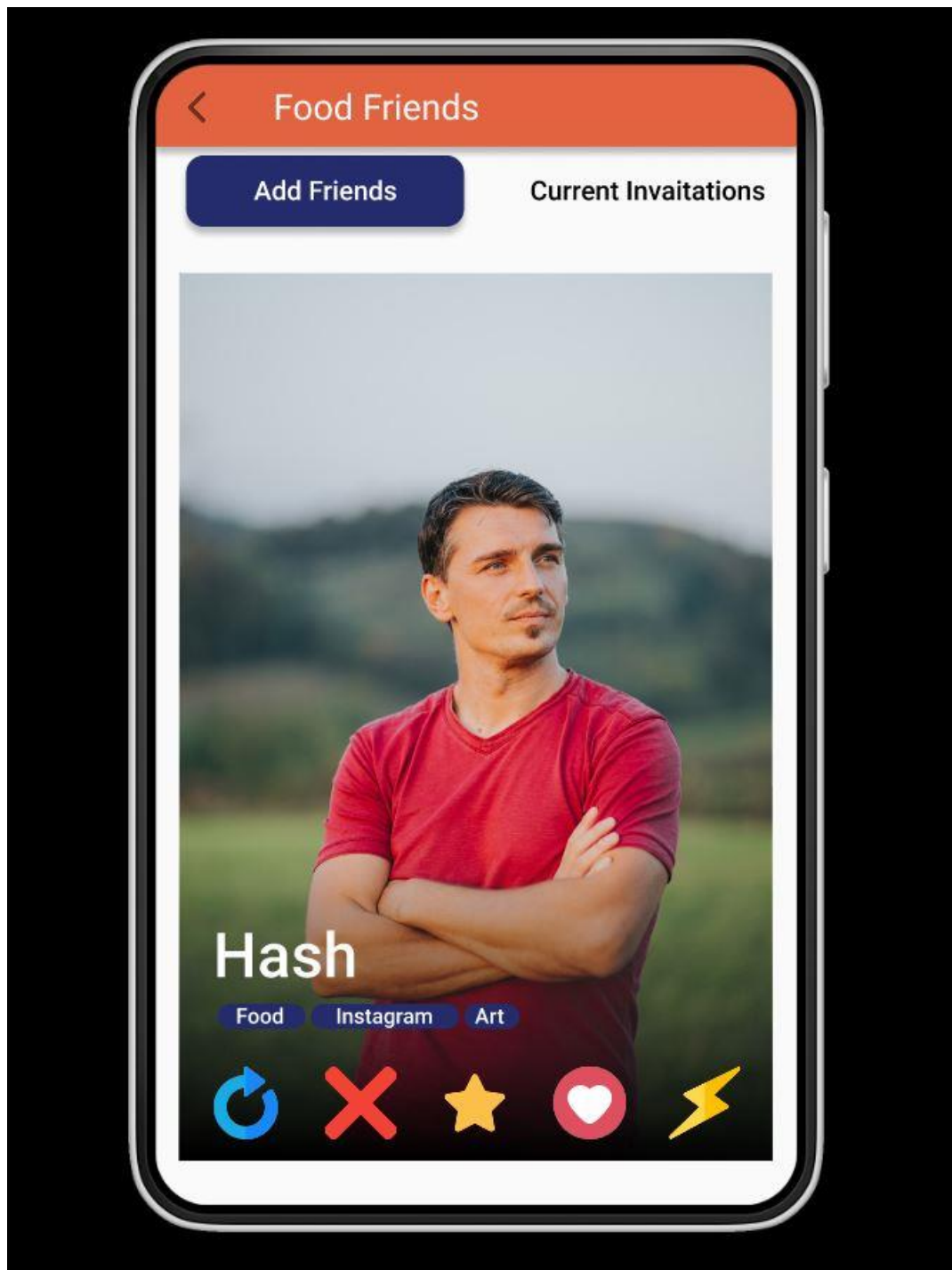


Figure 32: Customer app sent invitation page of Foodpatrons

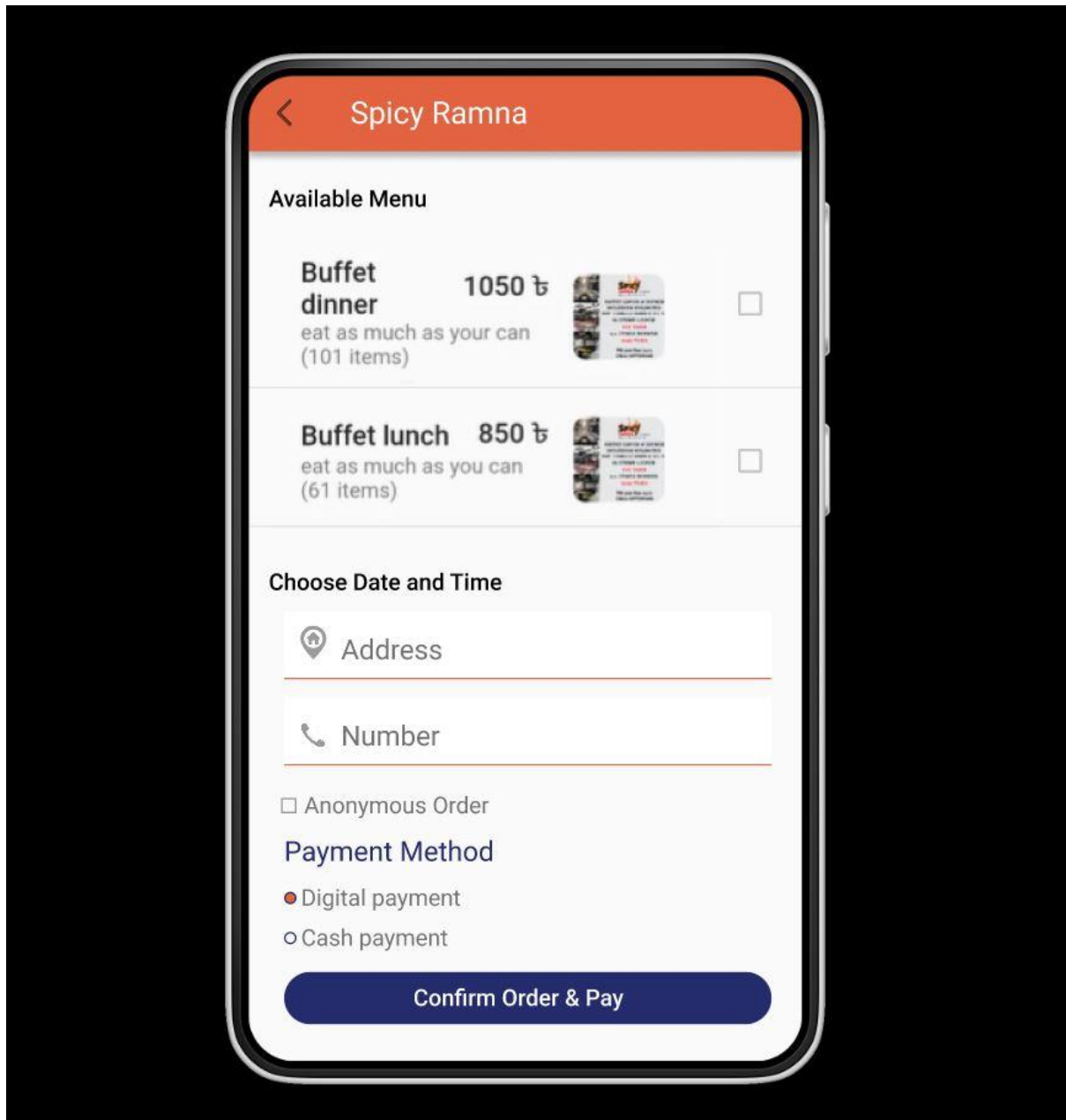


Figure 33: Customer app Food order form page of Foodpatrons

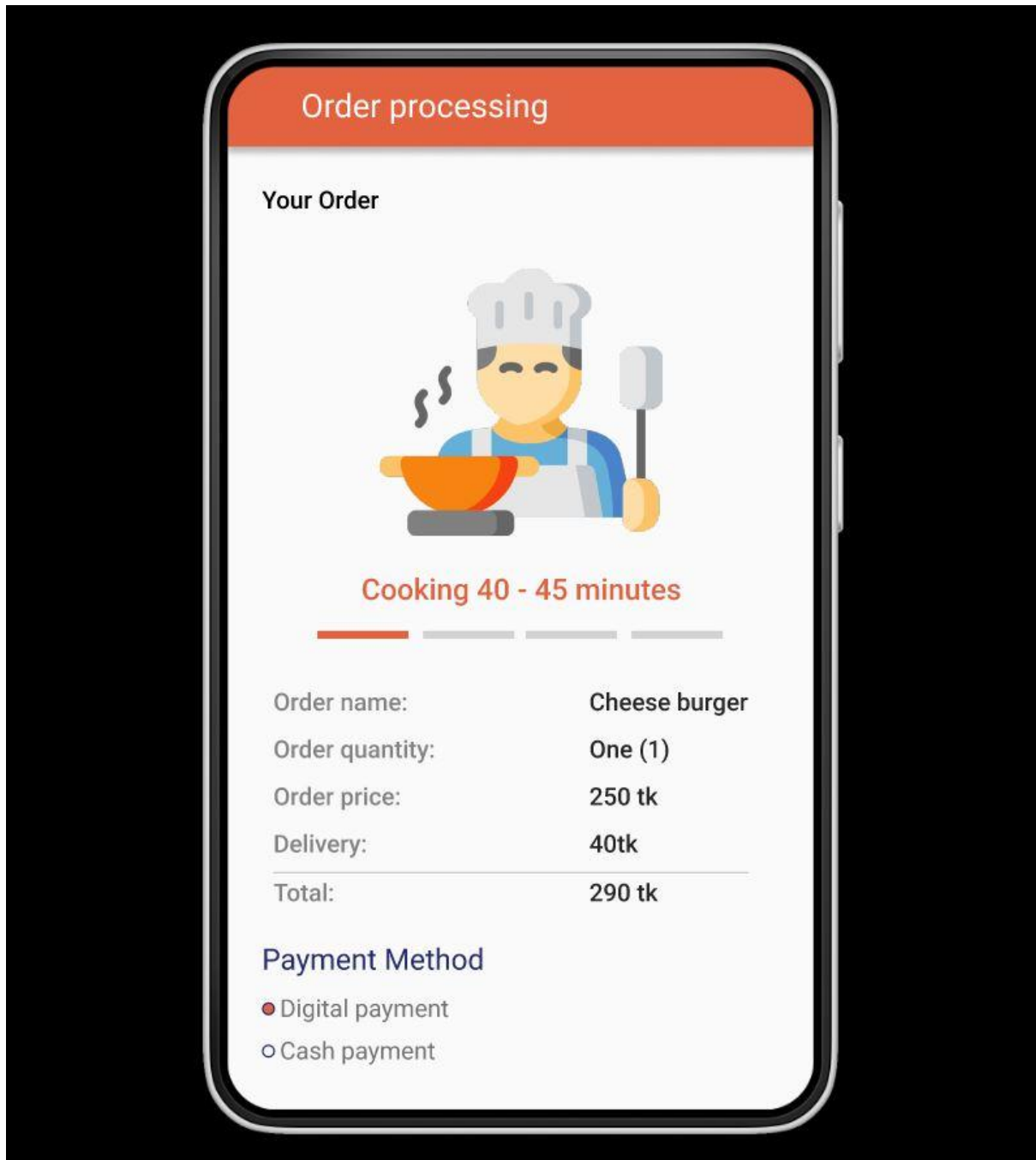


Figure 34: Customer app Food order Processing page (1) of Foodpatrons

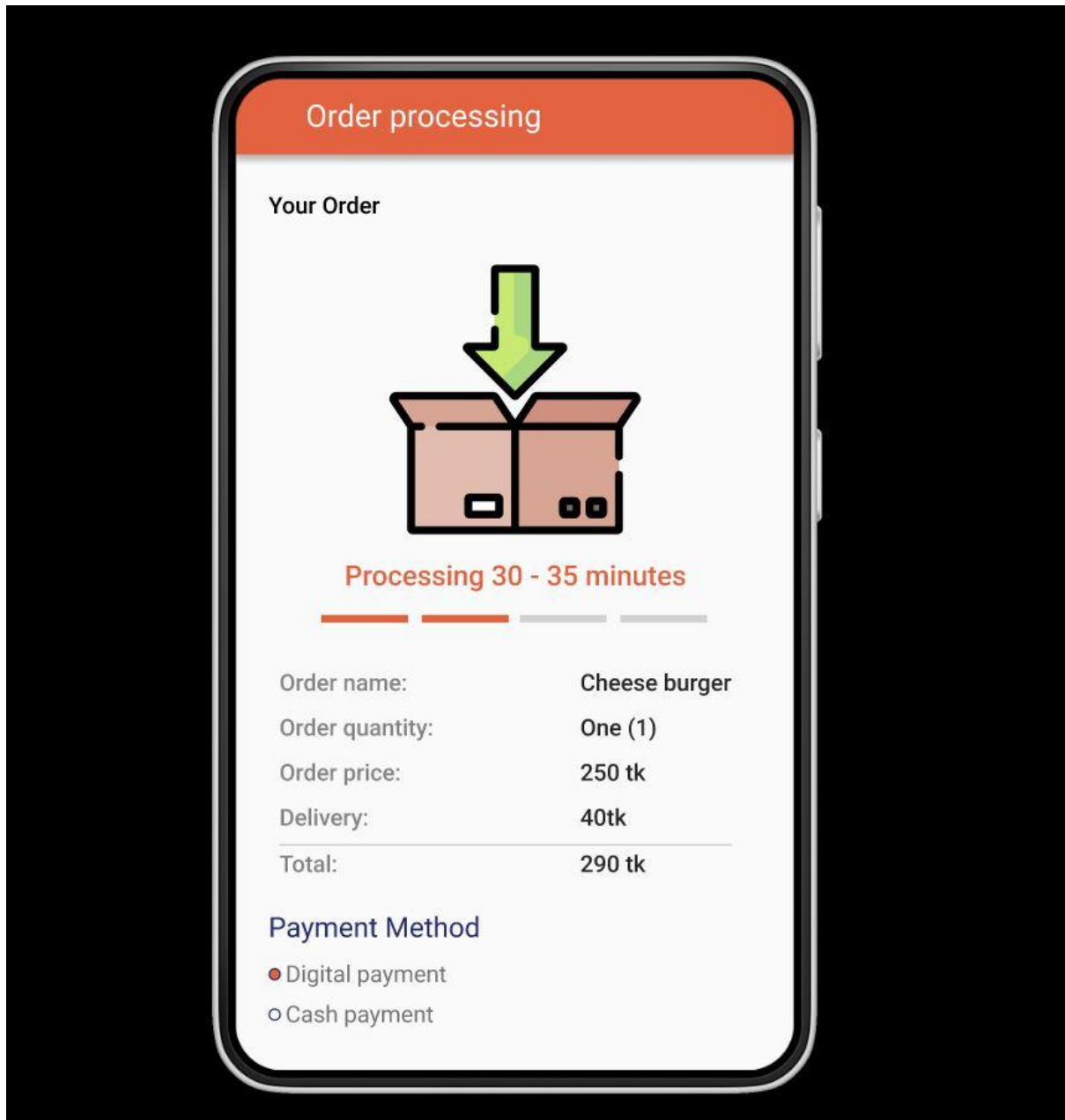


Figure 35: Customer app Food order Processing page (2) of Foodpatrons

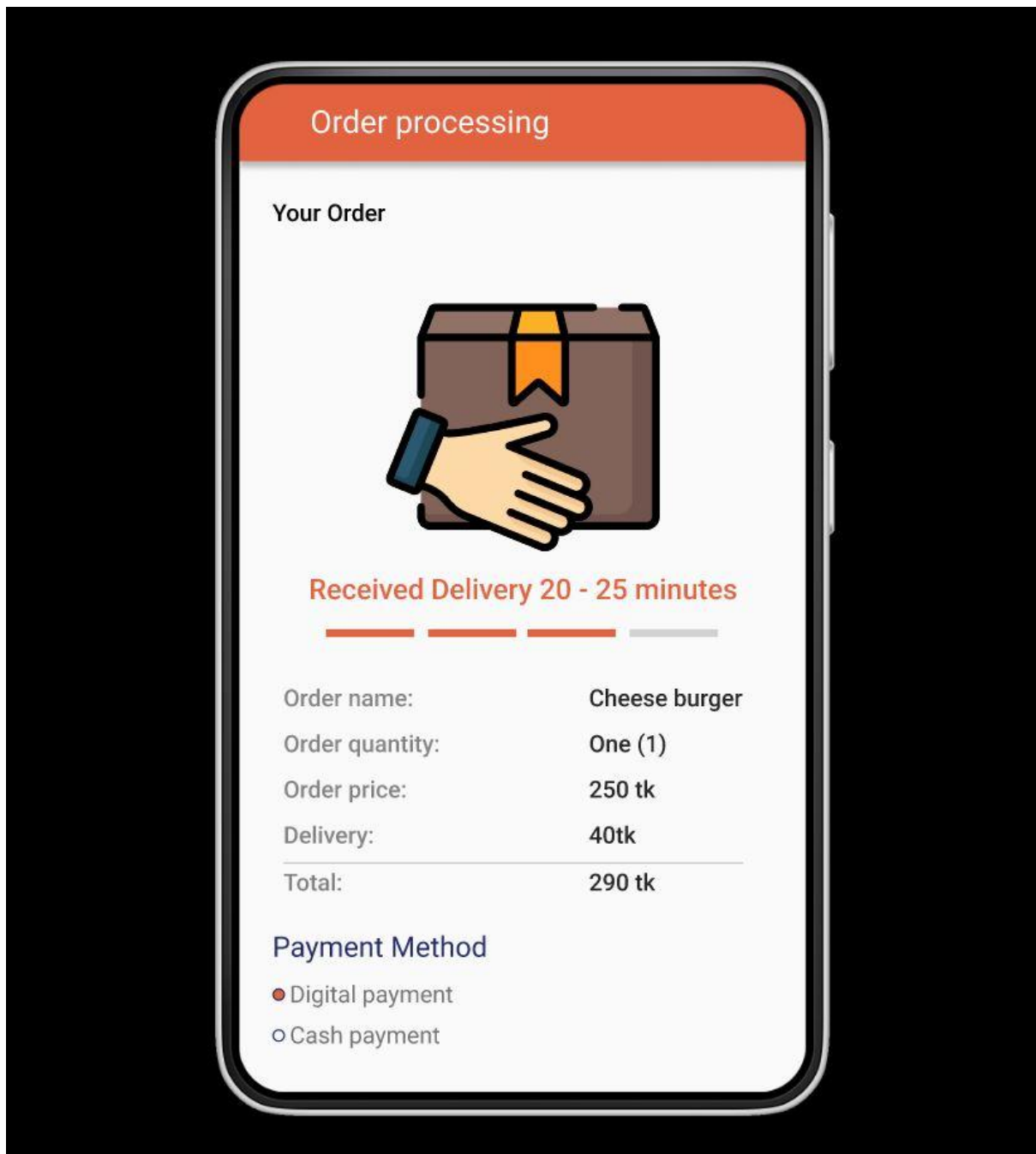


Figure 36: Customer app Food order Processing page (3) of Foodpatrons

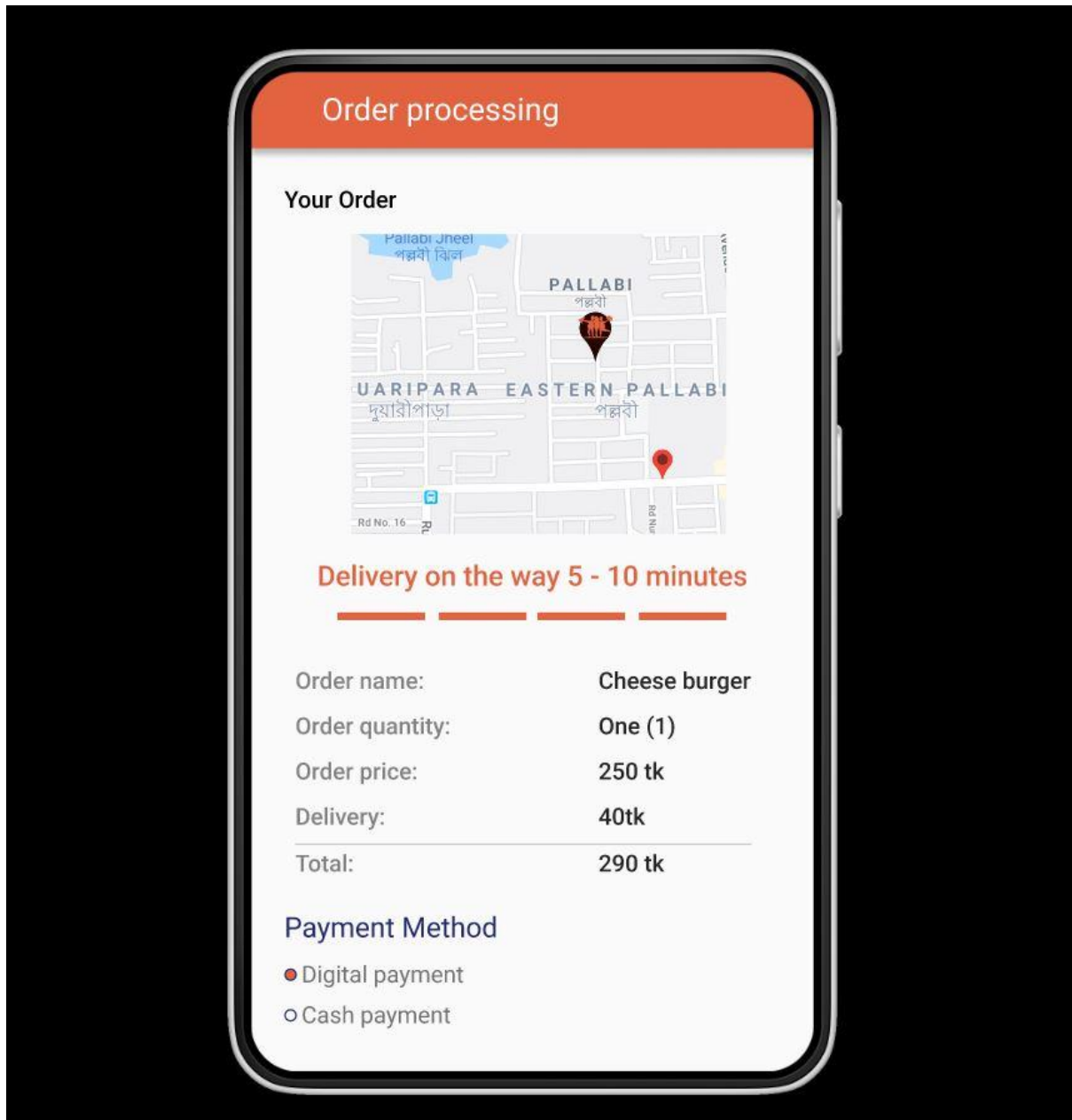


Figure 37: Customer app order current location tracking page of Foodpatrons

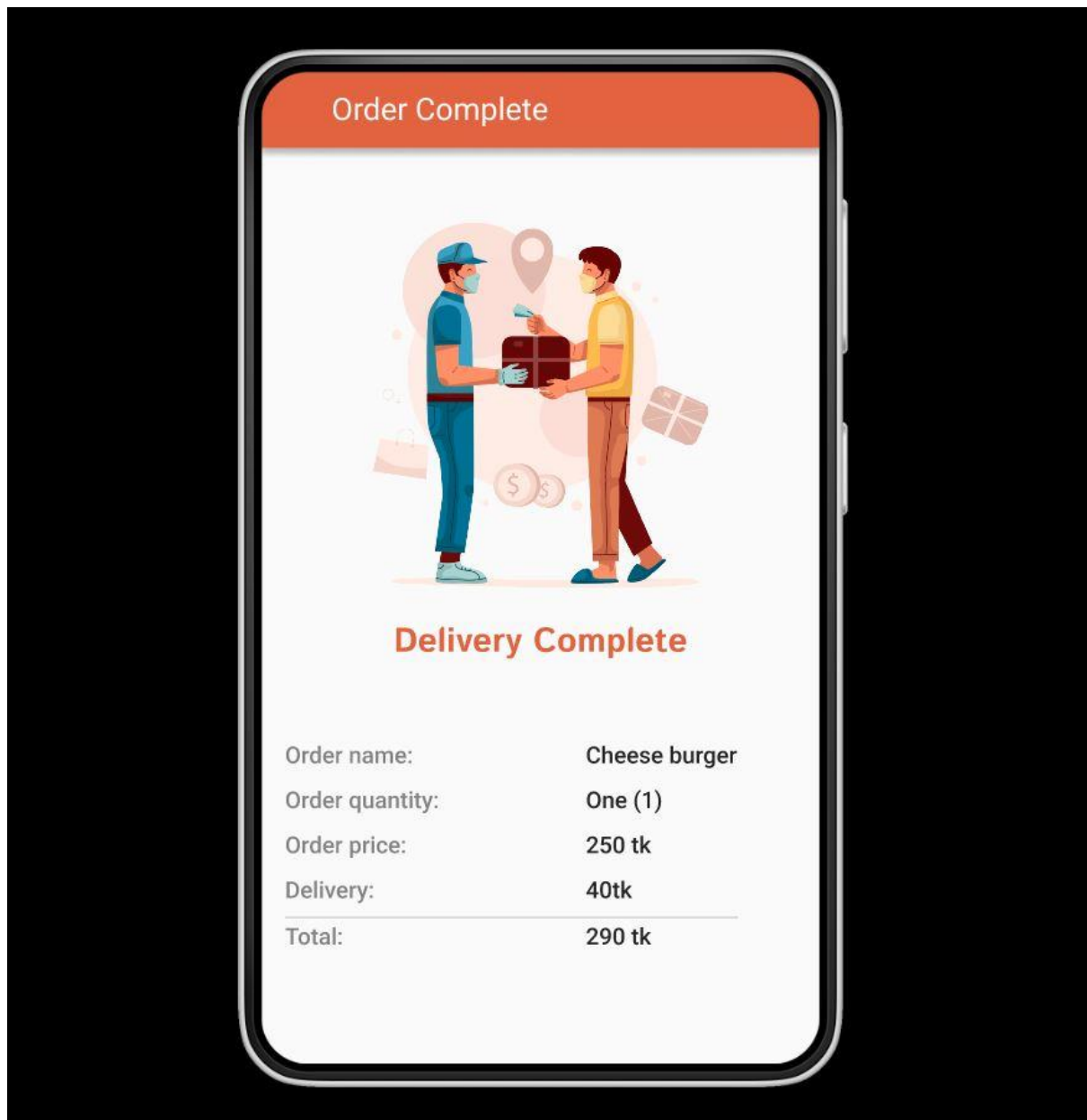


Figure 38: Customer app delivery complete page of Foodpatrons

Restaurant Owner app

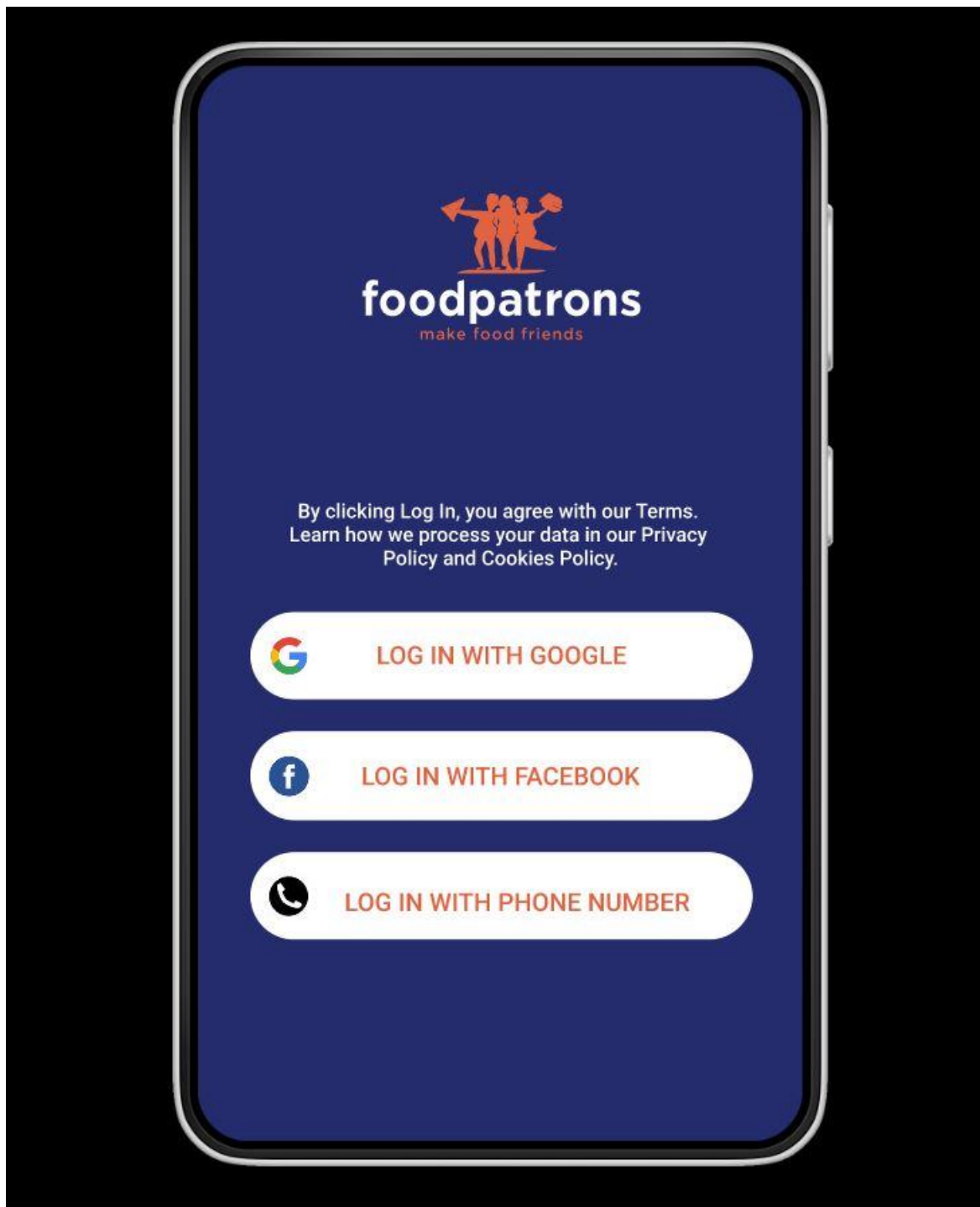


Figure 39: Restaurant Owner Login option page of Foodpatrons

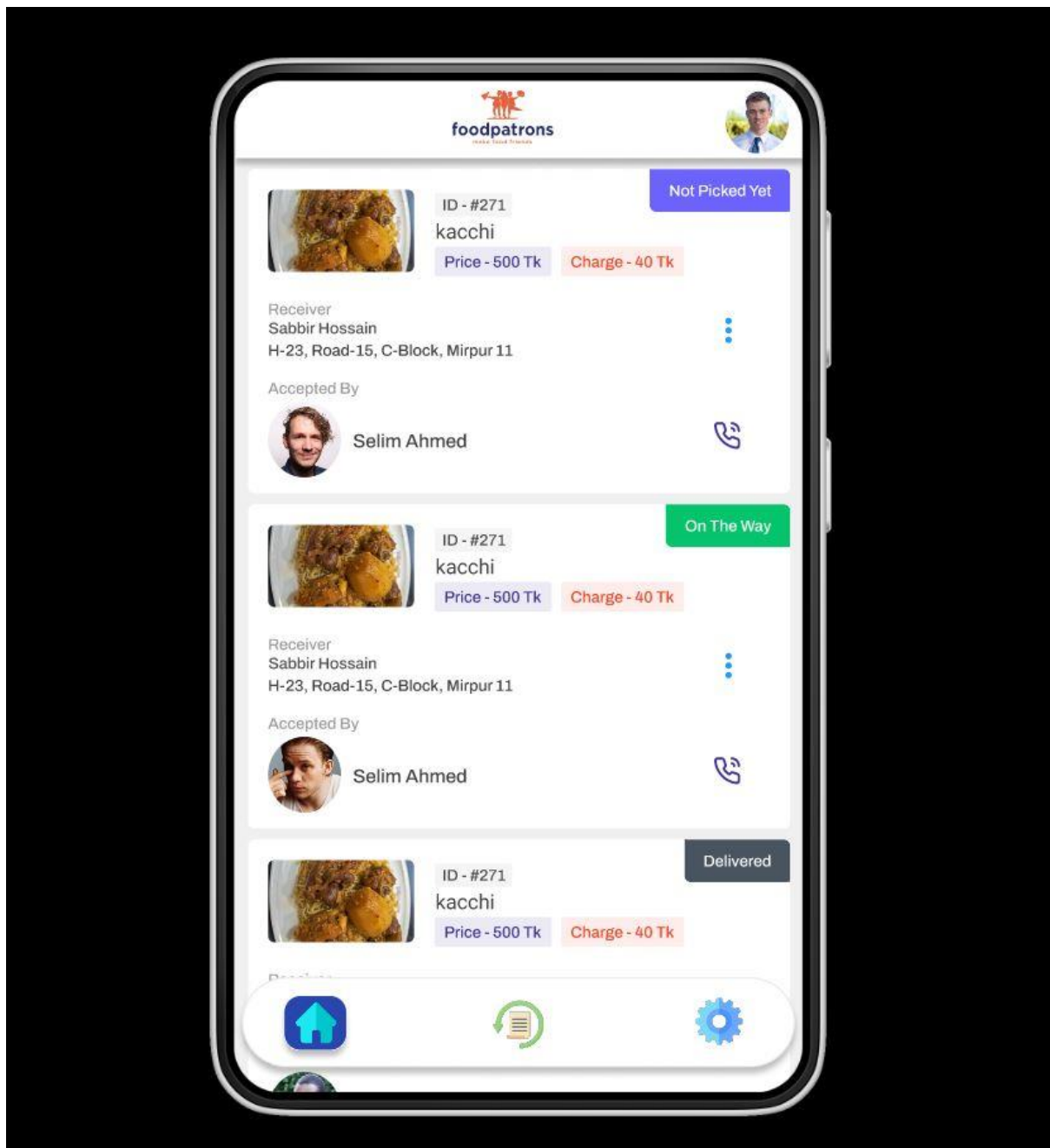


Figure 40: Restaurant Owner Home page of Foodpatrons

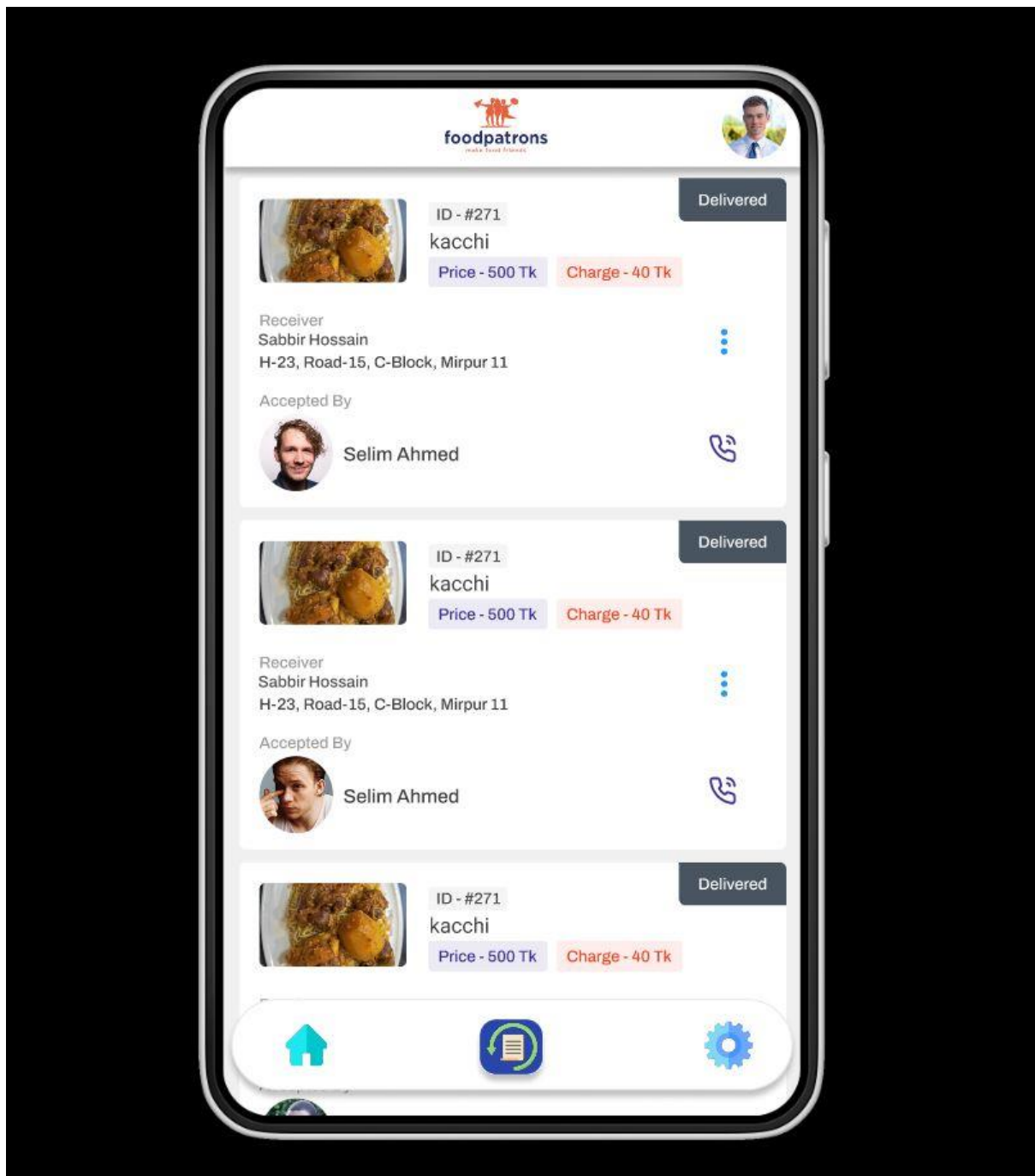


Figure 41: Restaurant Owner History page of Foodpatrons

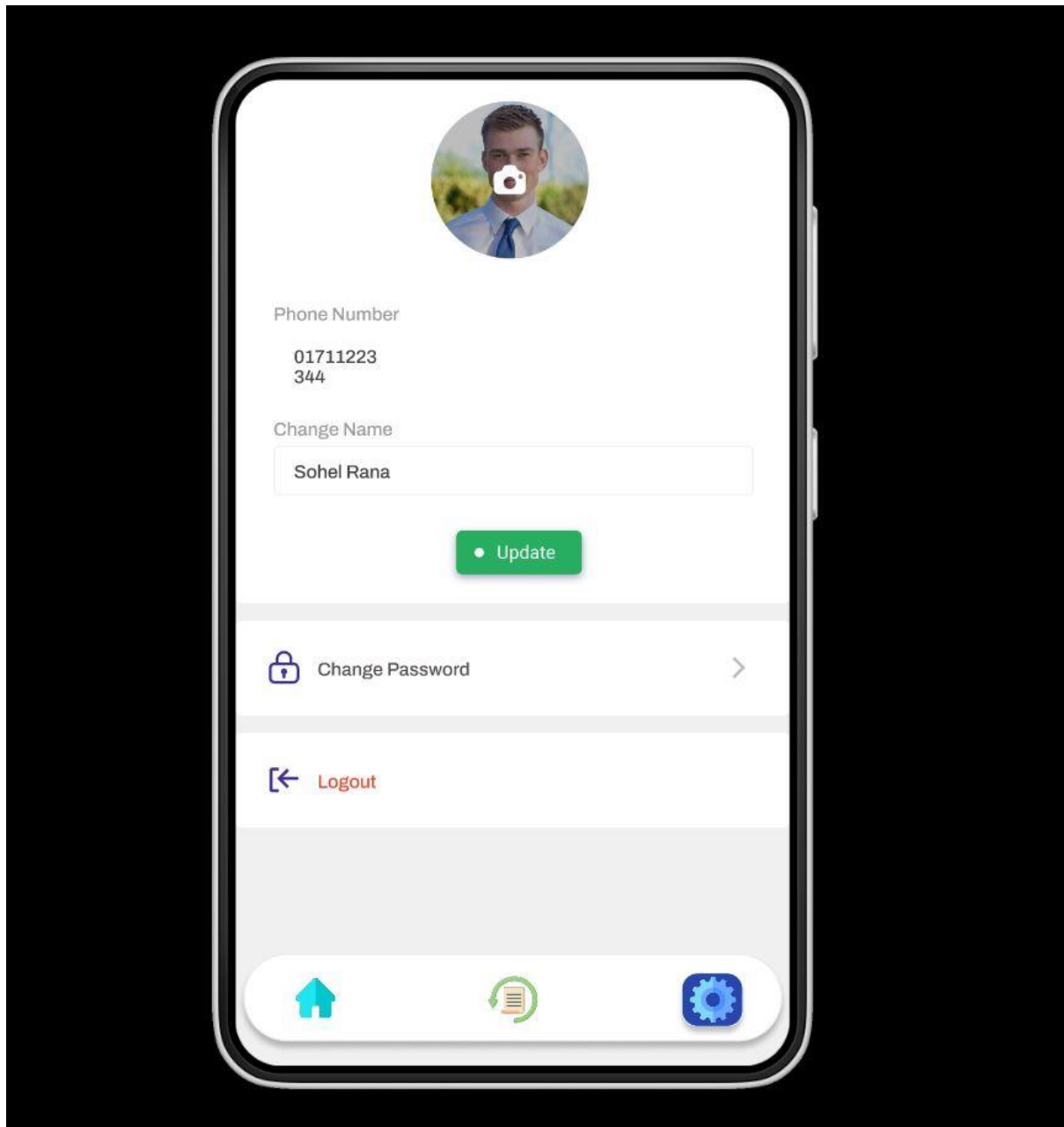


Figure 42: Restaurant Owner settings page of Foodpatrons

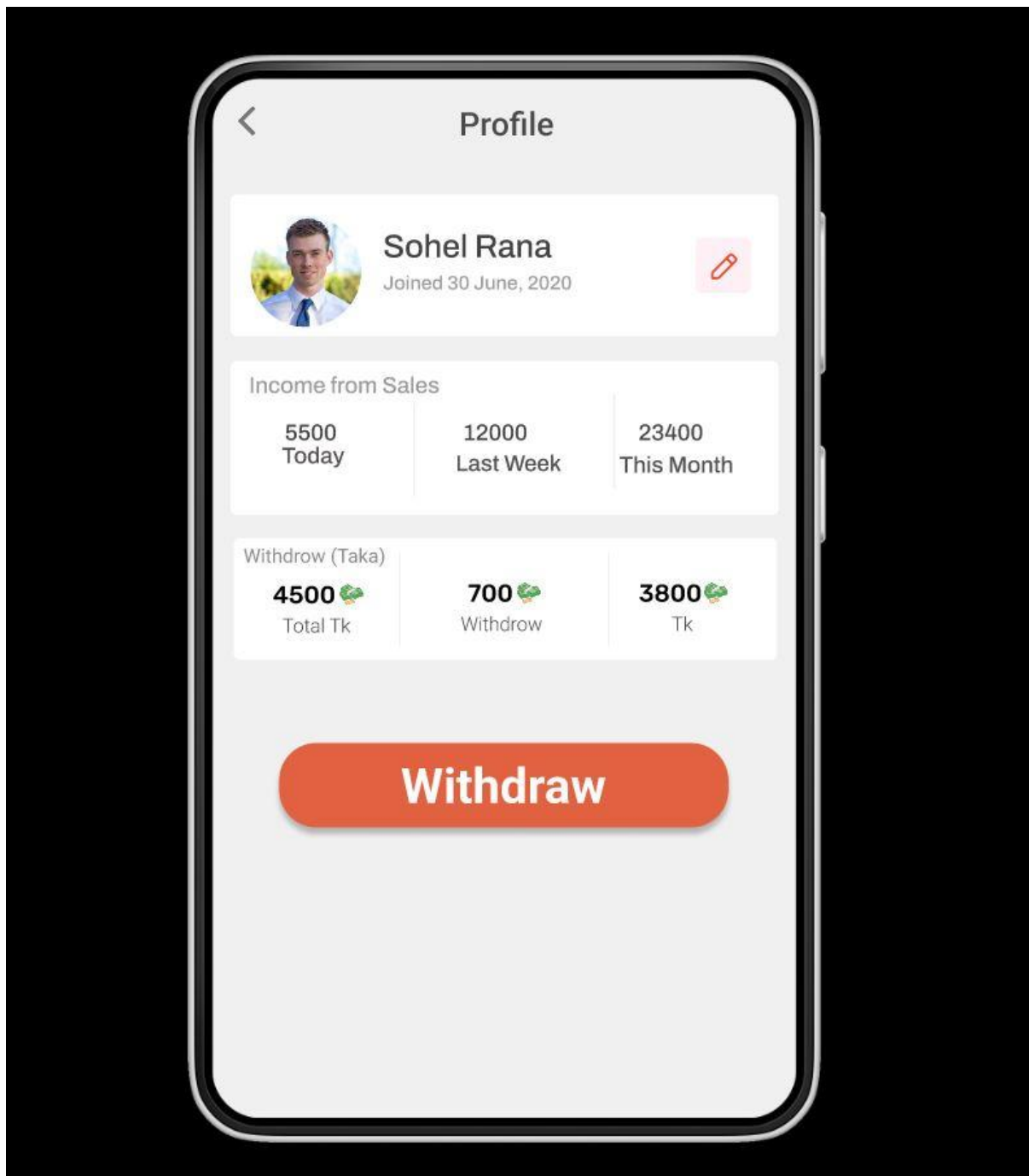


Figure 43: Restaurant Owner Profile page of Foodpatrons

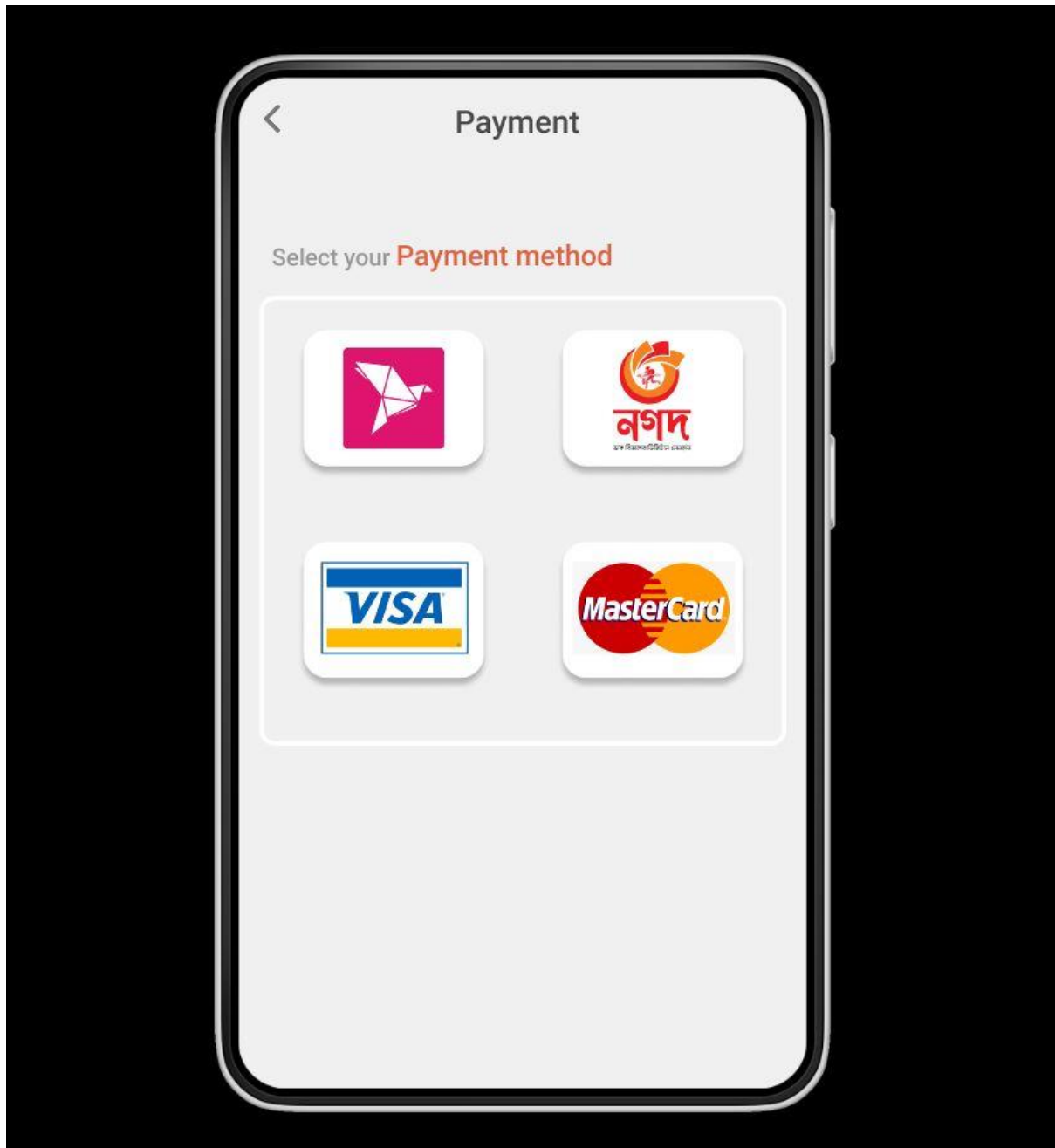


Figure 44: Restaurant Owner money withdraw option page of Foodpatrons

Delivery man app

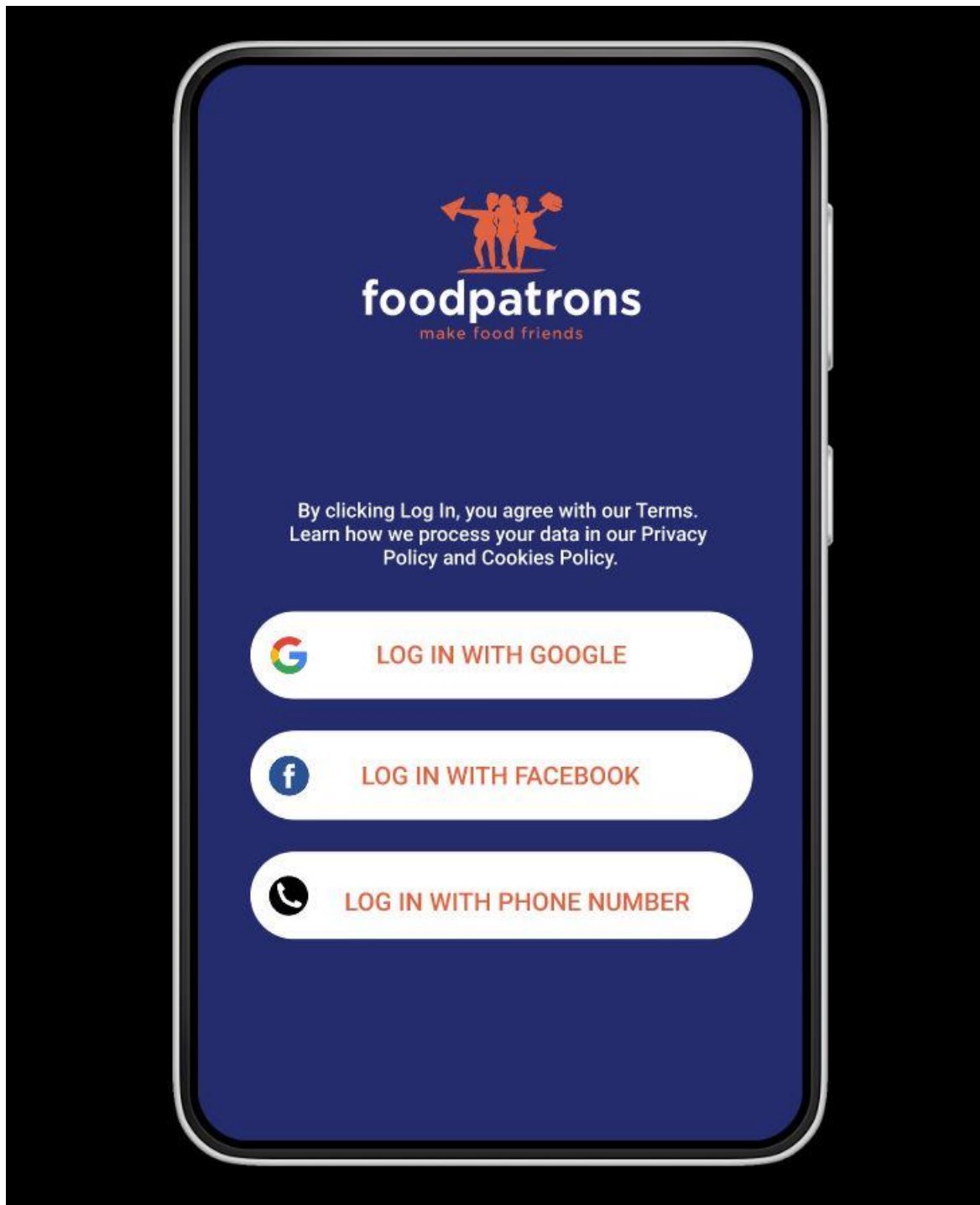


Figure 45: Delivery man Login option page of Foodpatrons

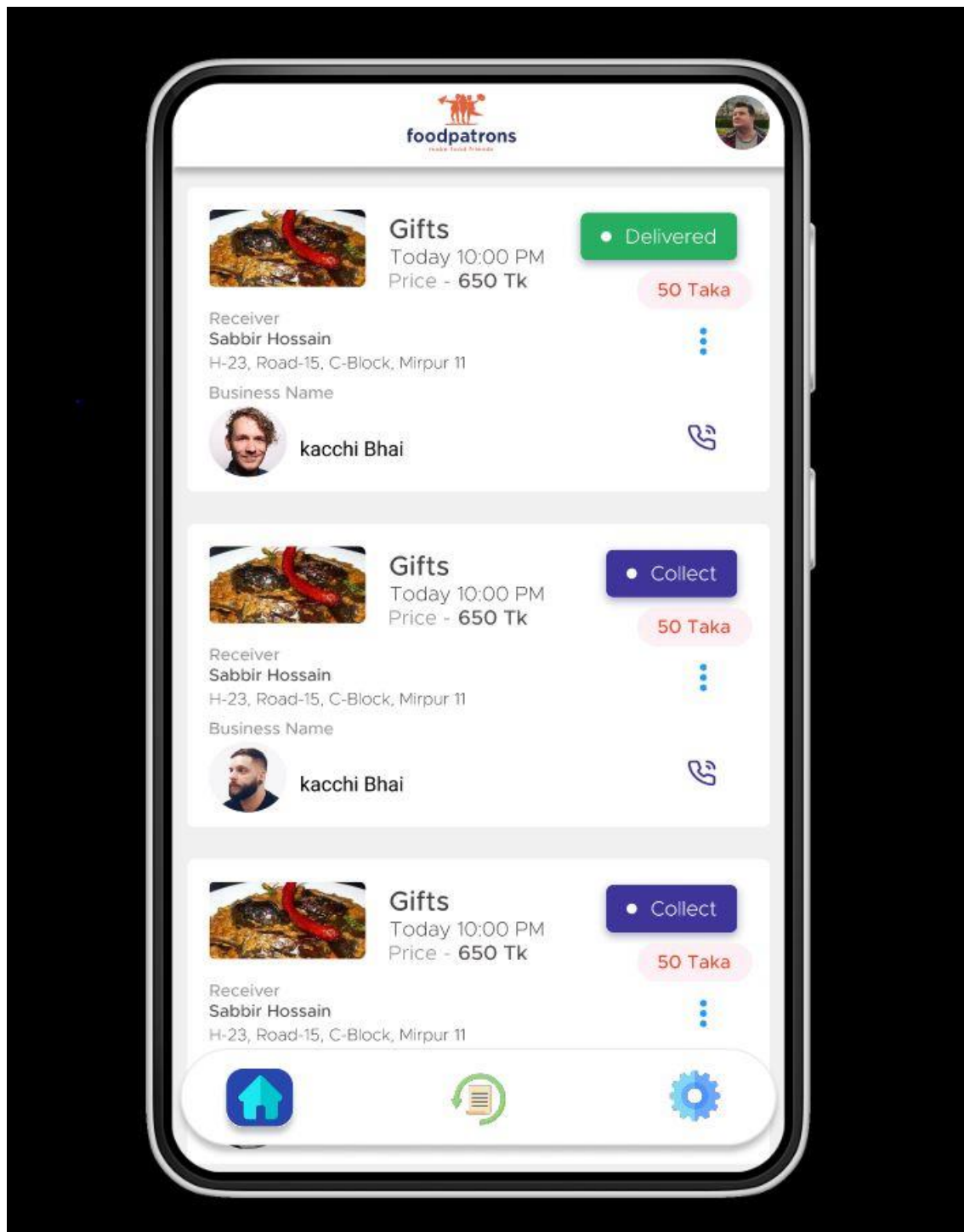


Figure 46: Delivery man home page of Foodpatrons

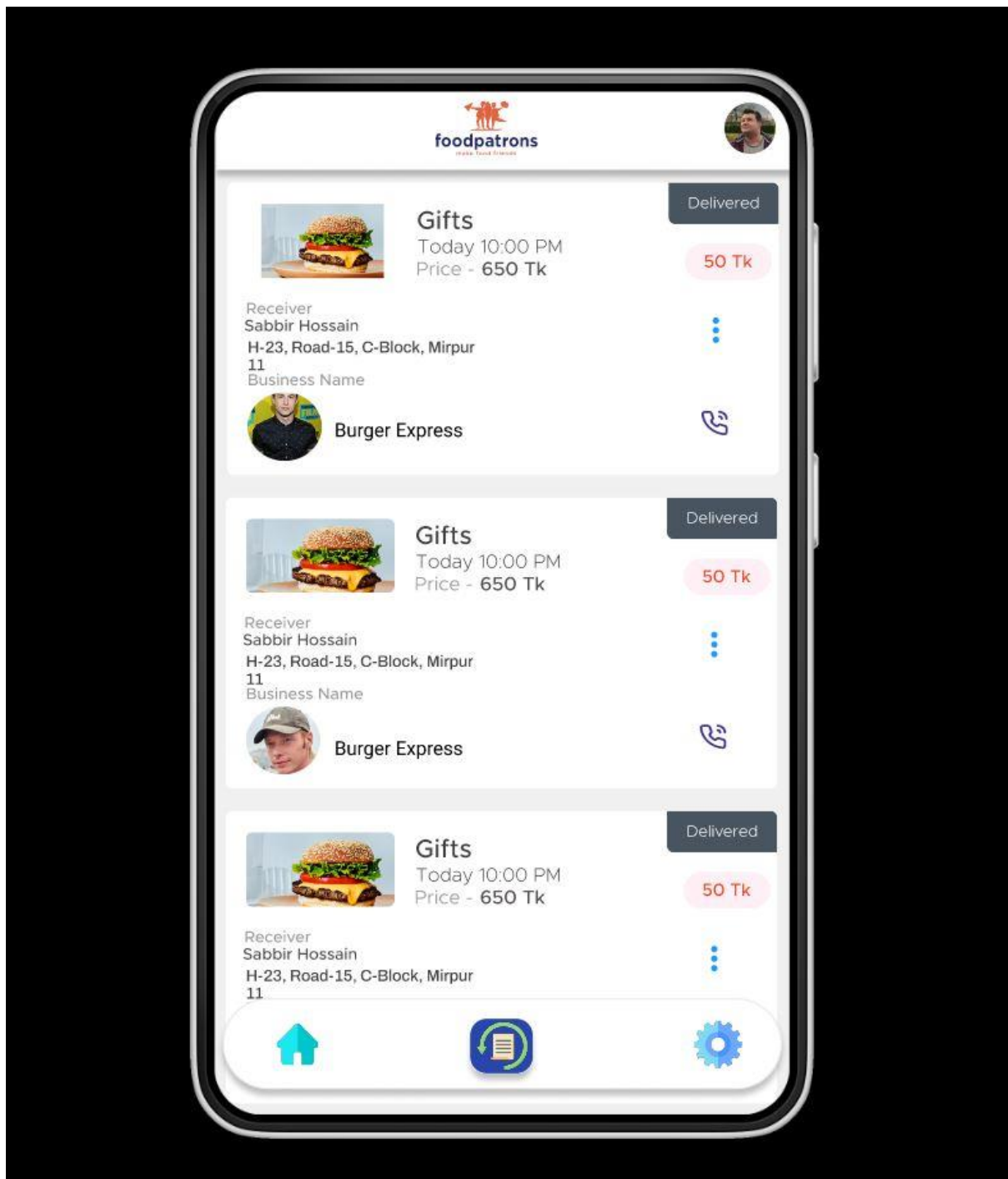


Figure 47: Delivery man history page of Foodpatrons

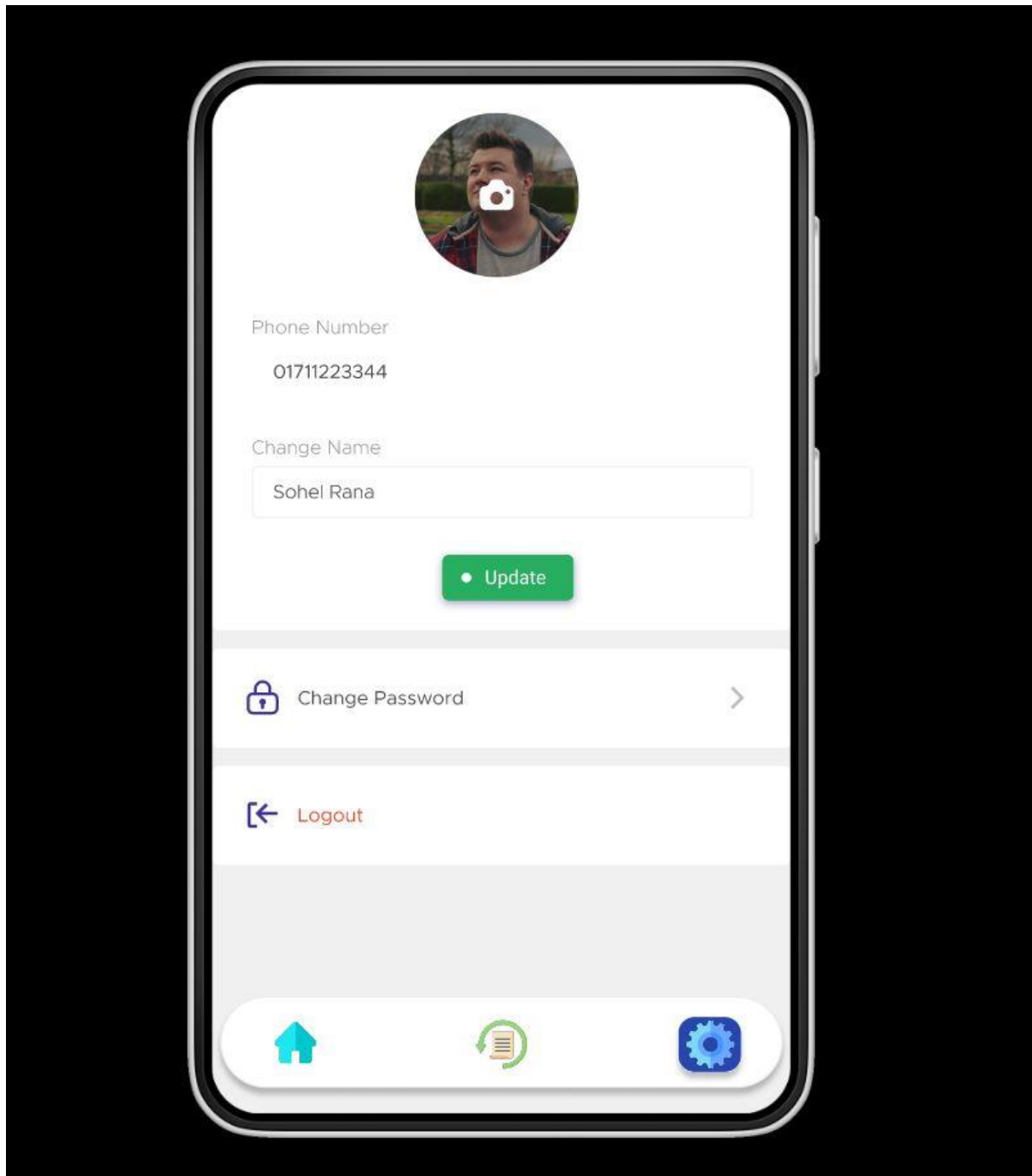


Figure 48: Delivery man settings page of Foodpatrons

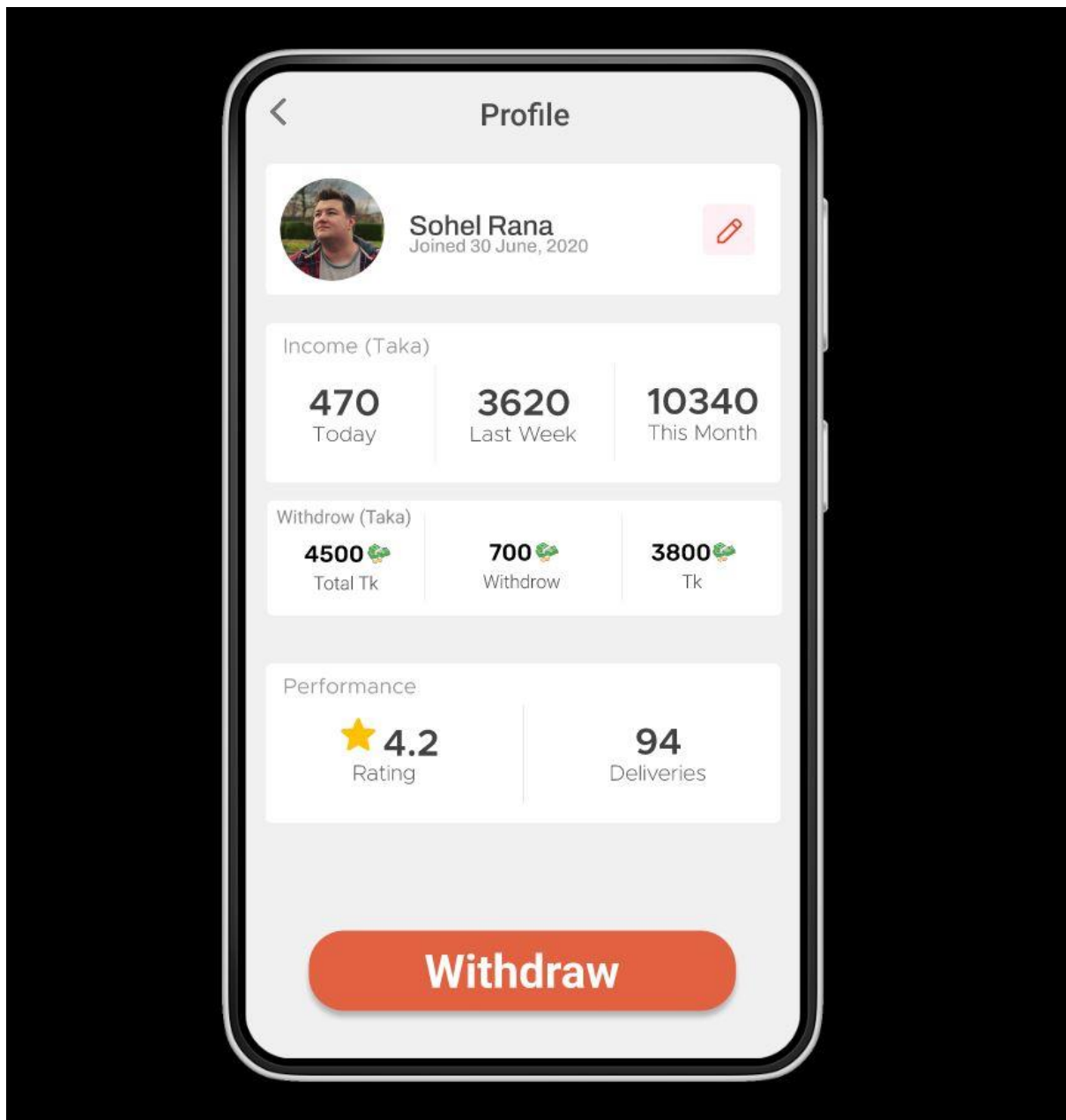


Figure 49: Delivery man profile page of Foodpatrons

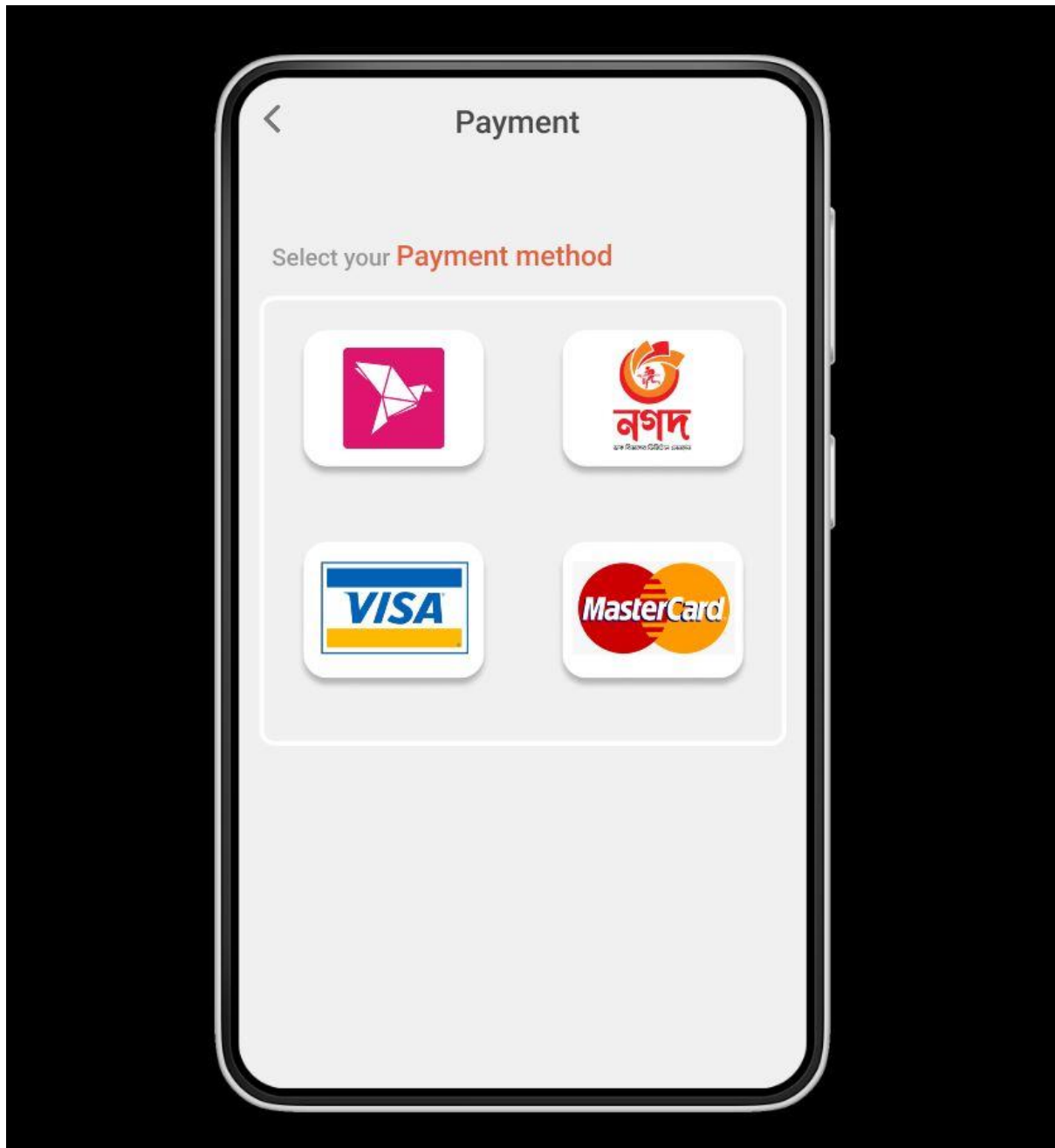


Figure 50: Delivery man withdraw method page of Foodpatrons

Chapter 9 – Engineering

9.1 New System Modules

The Foodpatrons system has a number of modules. However, the working method for several of the essential components is mentioned below.

Login Module

SL	User Action	SL	System Action
1	Admin, go to the login link and log in.	1	A login form will be displayed to the user.
2	Admin will provide login credentials	2	The system validates the given data.
3	Admin click on login button	3	The system verifies the administrator's identity. If you are authorized, you will be sent to the admin dashboard; otherwise, you will be sent back to the login page with an error notice.

Table 4: Login Module of Foodpatrons

Restaurant booking and order Module

SL	User Action	SL	System Action
1	Admin, can check the booking request.	1	The booking form will show
2	Admin can check order request.	2	The order details will show

Table 5: Restaurant booking and order Module of Foodpatrons

Delivery man location Module

SL	User Action	SL	System Action
1	Delivery man log in and home page	1	The system will ask for permission to use your current location.
2	Delivery man accept delivery	2	The system will locate and update the current position of the Delivery guy.

Table 6: Delivery man location Module of Foodpatrons

9.2 Use Case diagram of the Foodpatrons

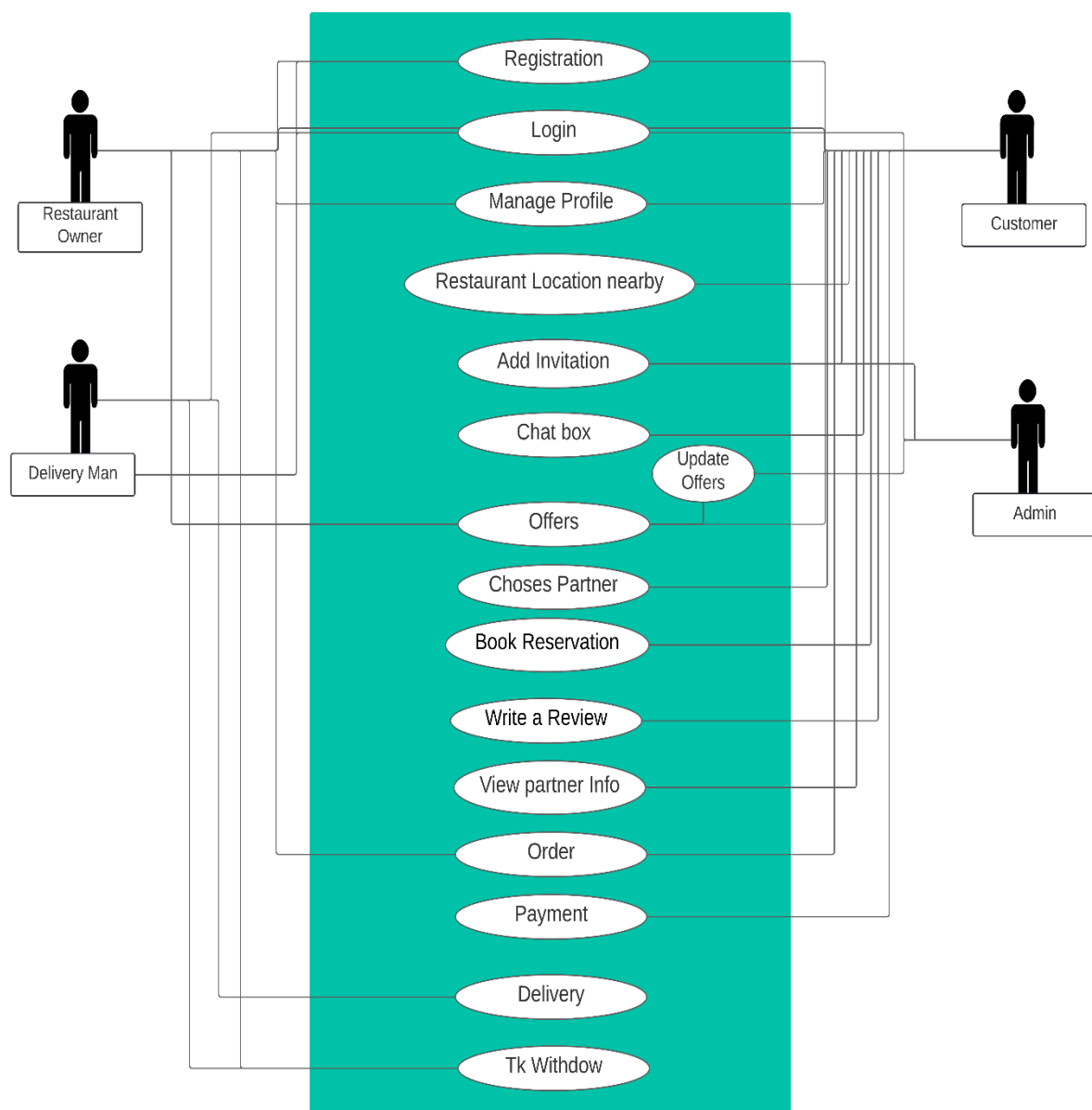


Figure 51: Use Case diagram of Foodpatrons

9.3 Peter Chen ERD Diagram

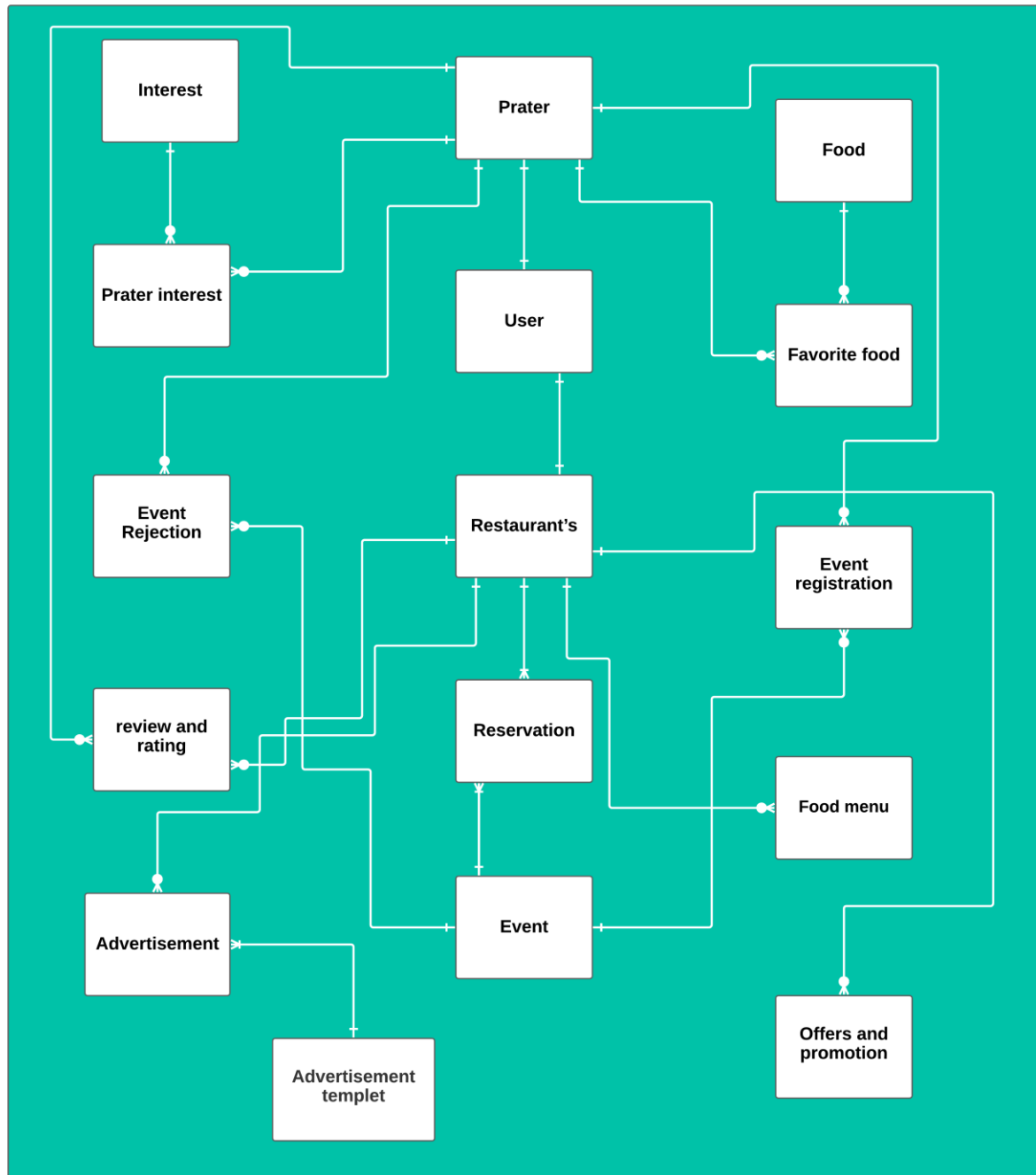


Figure 52: Entity Relationship Diagram of Foodpatrons

9.4 System Interface Hand drawing Design / Prototype

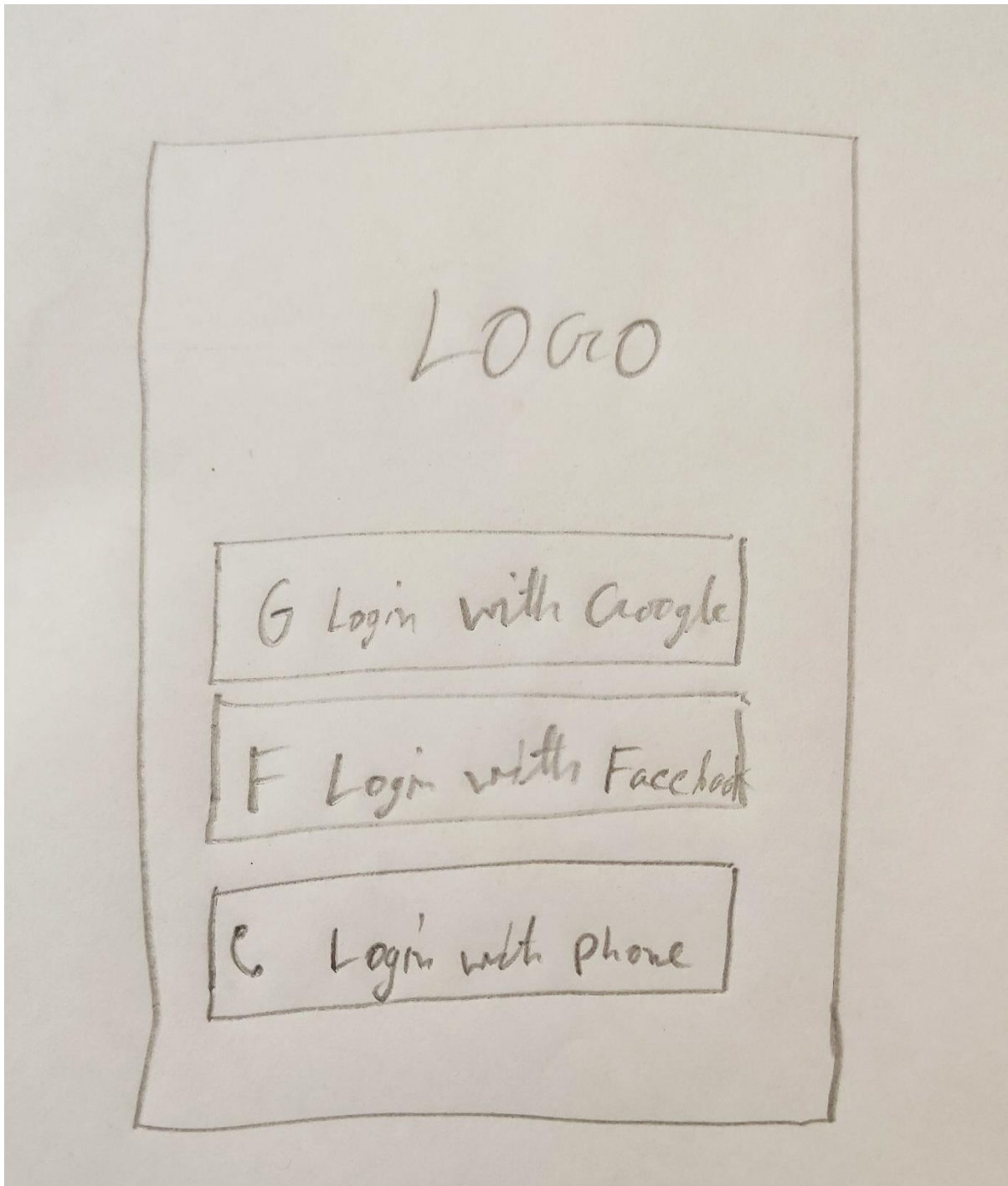


Figure 53: Customer login page

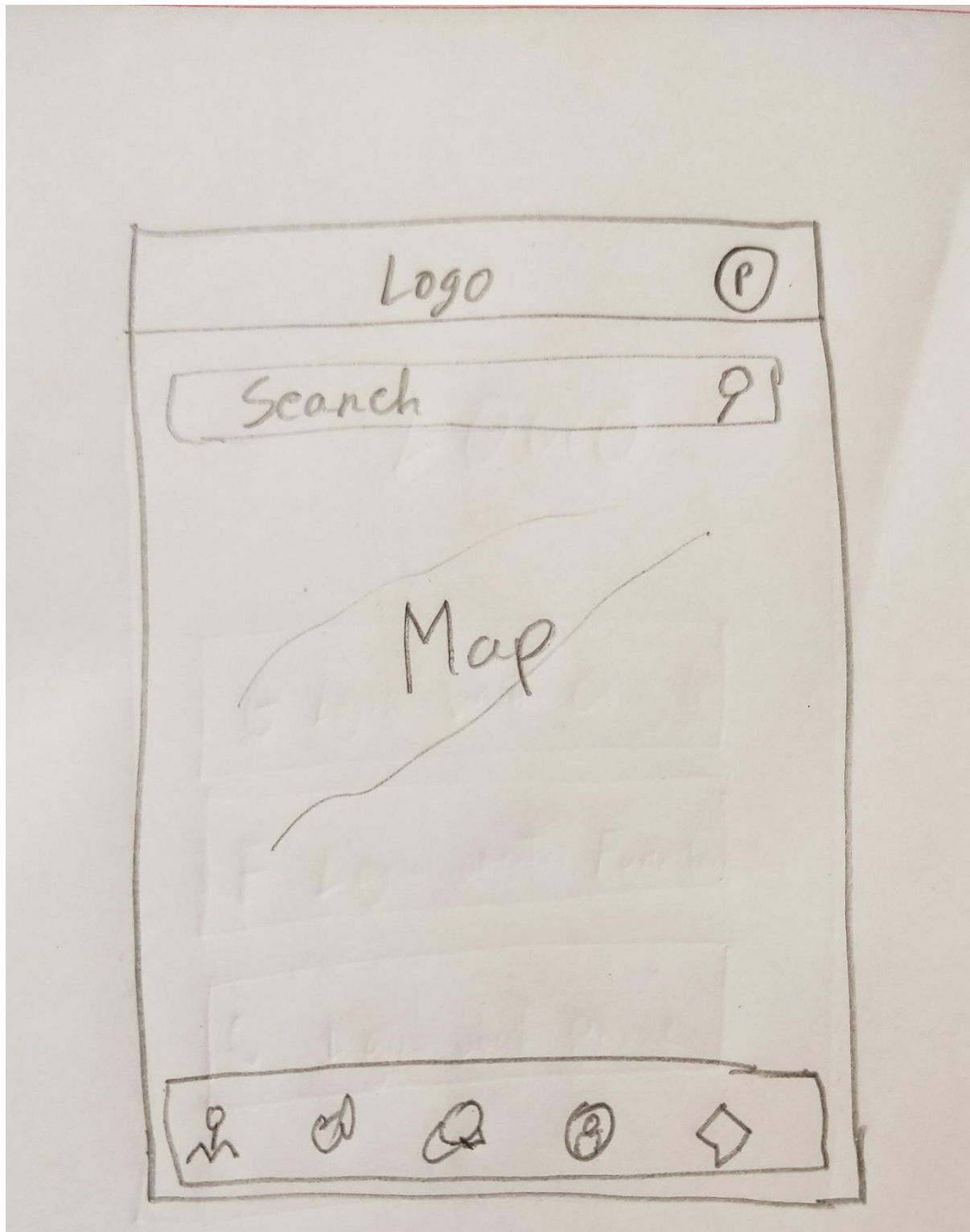


Figure 54: Customer Home page

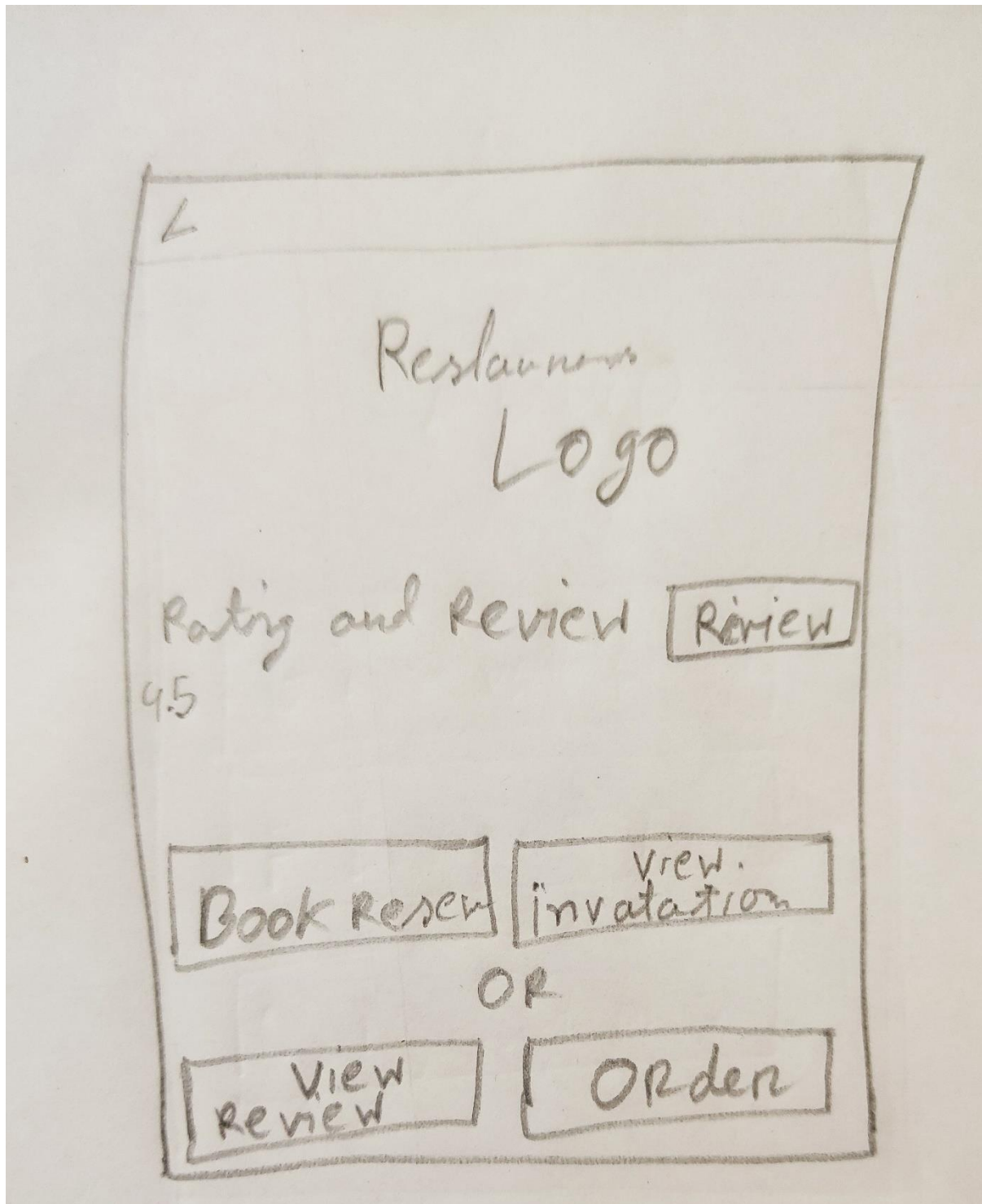


Figure 55: Restaurant's

A hand-drawn form for a book reservation system. The form is enclosed in a large rectangular border. At the top, there is a section labeled 'Restaurant Name' with a left-pointing arrow. Below this is a section labeled 'Available Menu'. Underneath the menu section is a large rectangular box labeled 'Menu'. Below the menu box are three separate rectangular boxes, each containing a label: 'Time & Duration', 'Number of friends', and 'Book Reservation'.

← Restaurant Name

Available Menu

Menu

Time & Duration

Number of friends

Book Reservation

Figure 56: Book Reservation

A hand-drawn sketch of a form titled 'Create Invitation'. The form is enclosed in a large rectangular border. At the top, the title 'Create Invitation' is written in a cursive font. Below the title, there are four input fields, each represented by a horizontal rectangle. The first field contains the word 'City' and has a small checkmark '✓' at its right end. The second field contains the word 'Restaurant'. The third field contains the word 'Budget'. The fourth field contains the word 'Time'. At the bottom of the form, there is a button labeled 'Create Invitation'.

Figure 57: Create Invitation

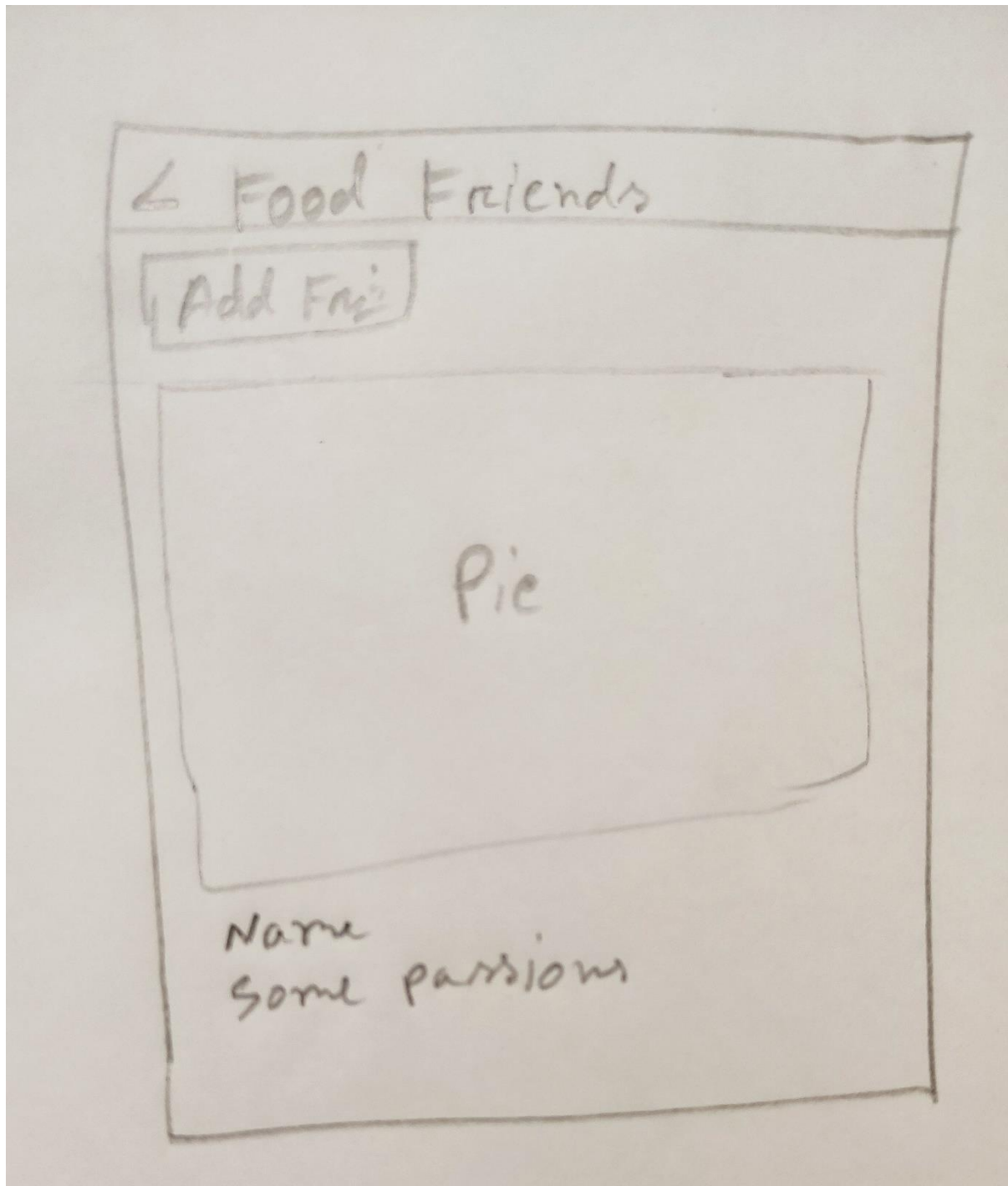


Figure 58: Food friend

A hand-drawn sketch of a profile form. The form is enclosed in a large rectangular border. At the top, the word "profile" is written in a cursive script. Below this, there are four distinct sections, each in its own rectangular box. The first box contains a circular icon with a stylized 'a' and the word "Name" written next to it. The second box is labeled "Income (Tk)" and contains a horizontal line with two vertical tick marks. The third box is labeled "withdraw (Tk)" and also contains a horizontal line with two vertical tick marks. The fourth box, at the bottom, is labeled "withdraw" and is slightly offset to the right.

Figure 59: Delivery man or Restaurant owner profile

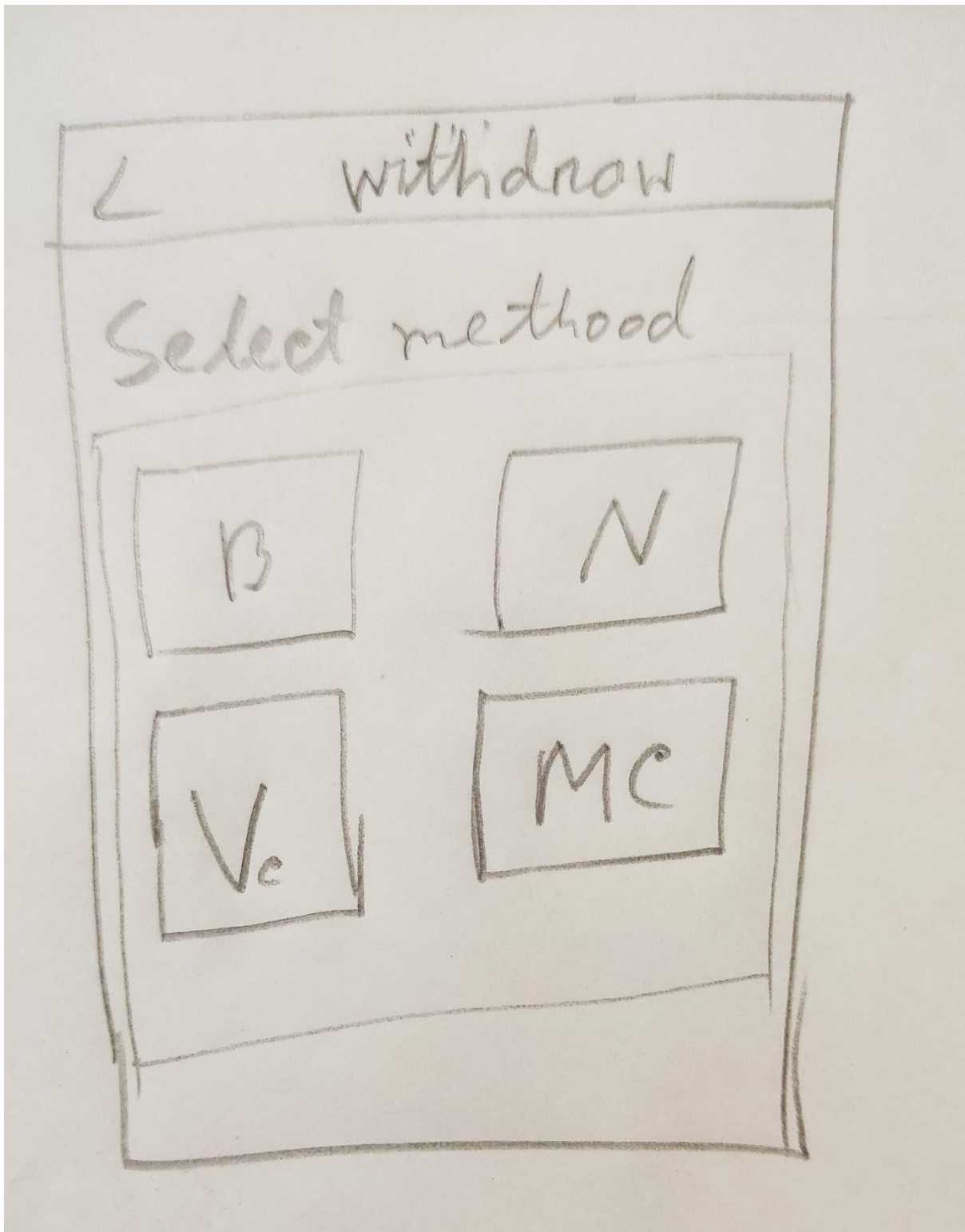


Figure 60: Money withdraw

9.5 System Interface Design / Prototype

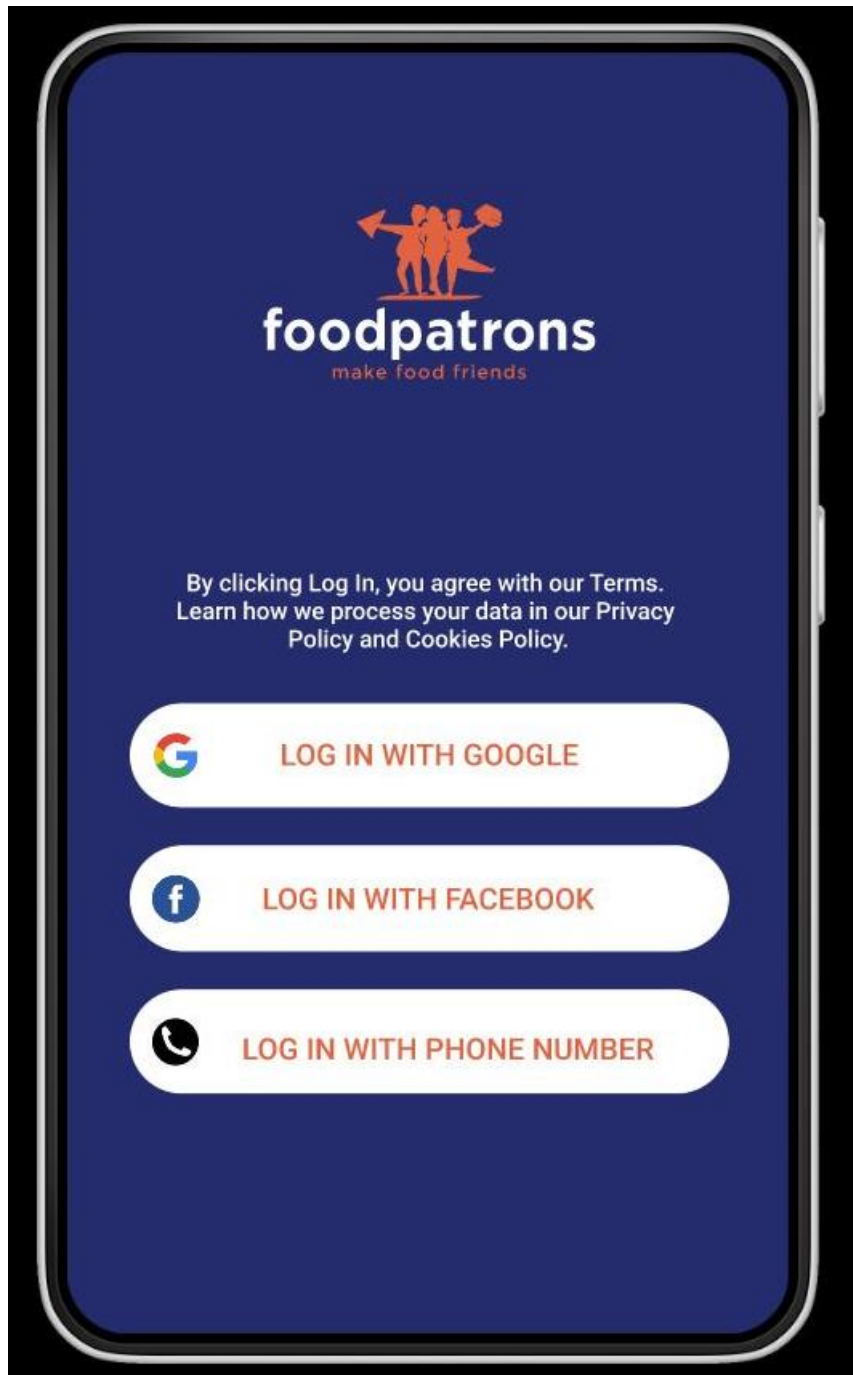


Figure 61: Customer app Login option page of Foodpatrons

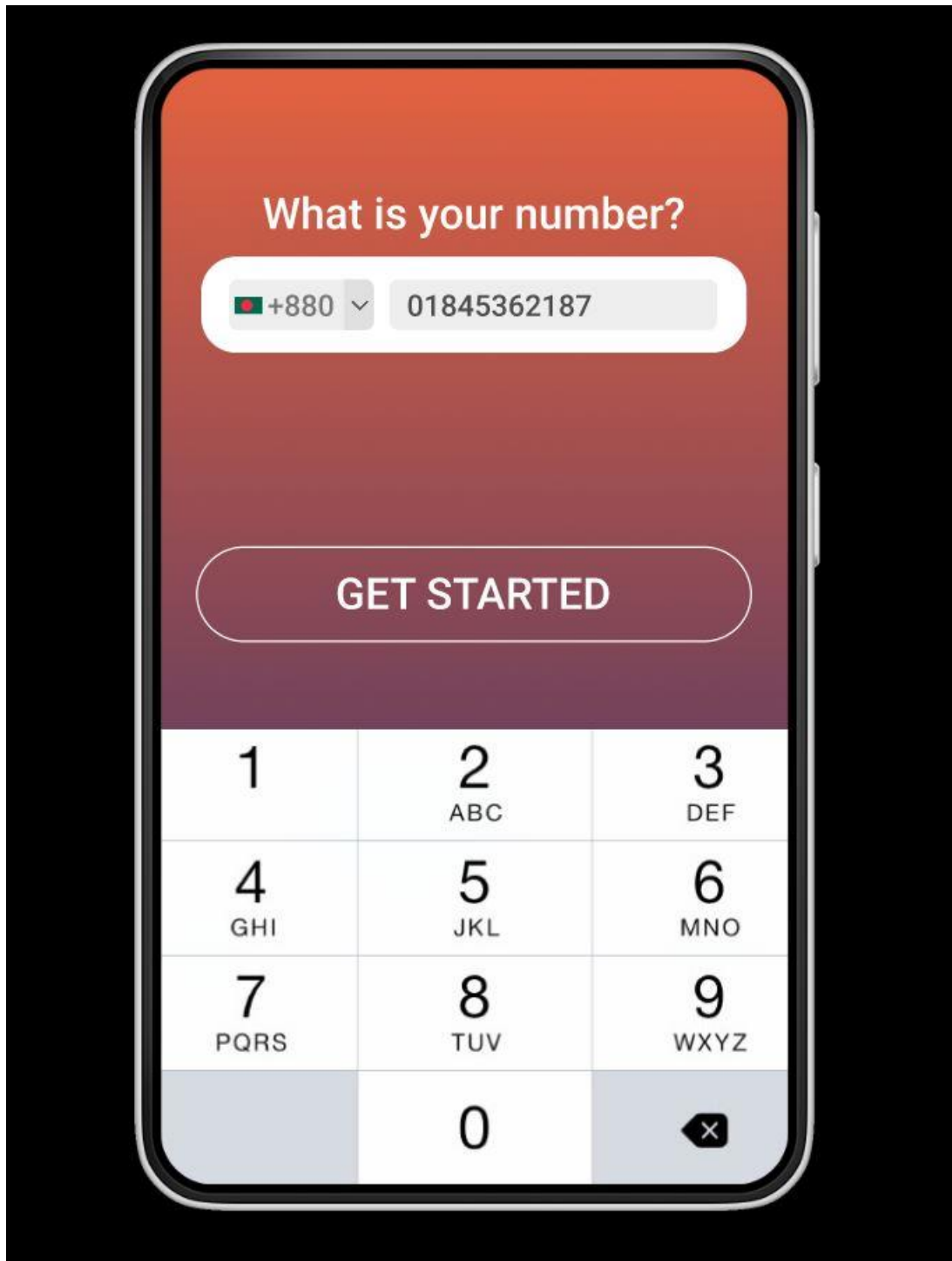


Figure 62: Customer app login with phone number page of Foodpatrons

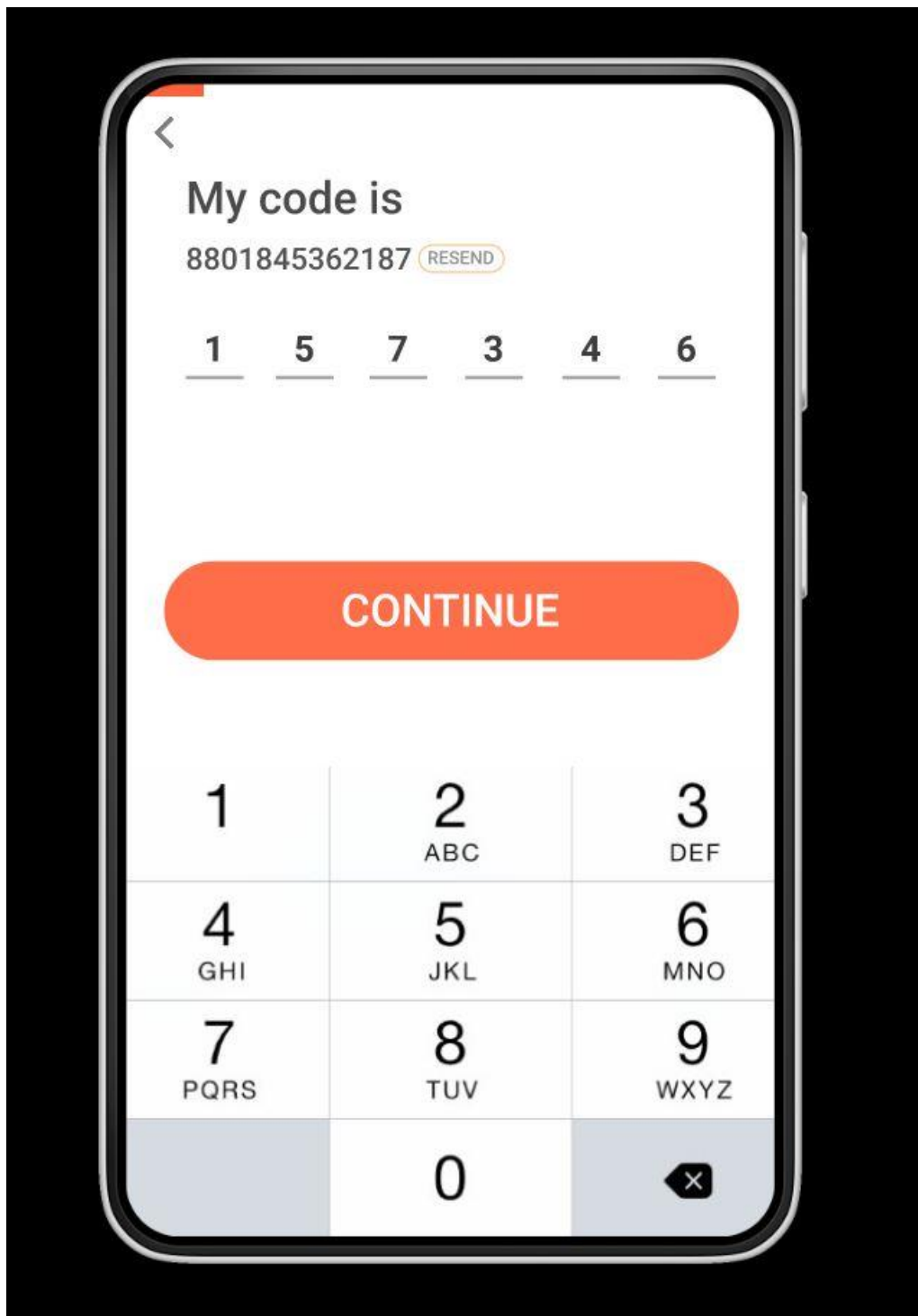


Figure 63: Customer app OTP page of Foodpatrons

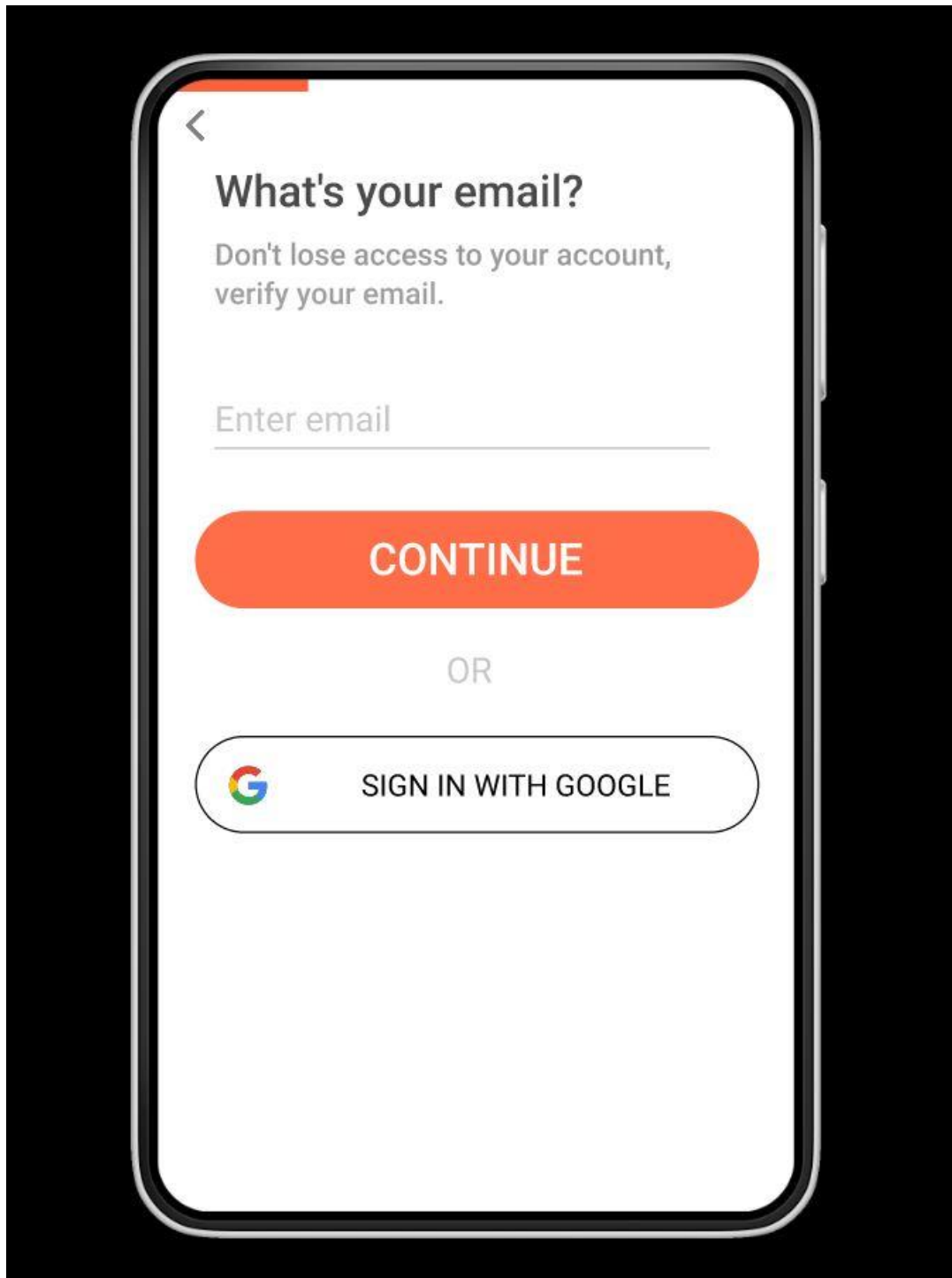


Figure 64: Customer app Email entry page of Foodpatrons

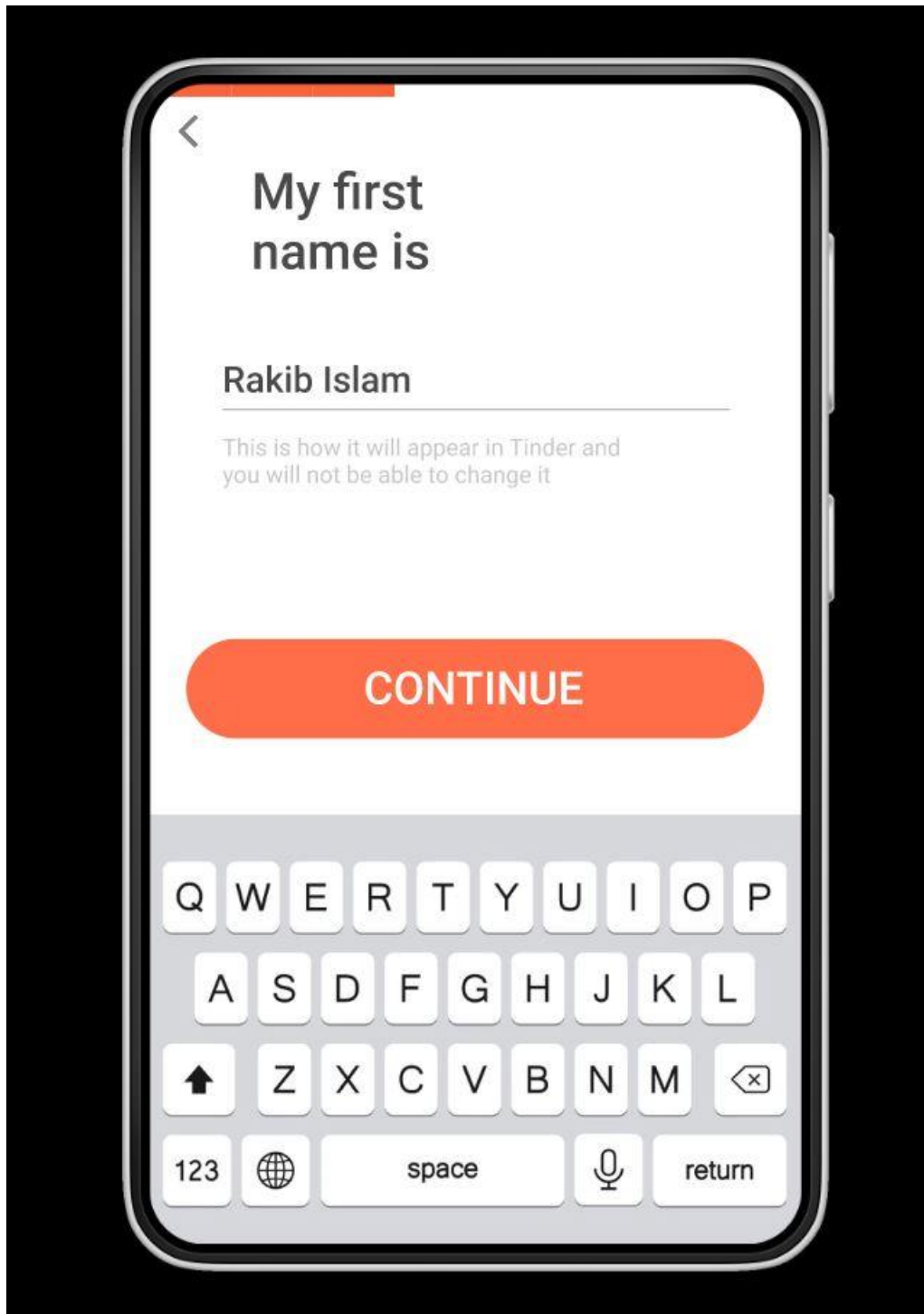


Figure 65: Customer app Name entry page of Foodpatrons

<

My birthday is

M M / D D / Y Y Y Y

Your age will be public

CONTINUE

1	2 ABC	3 DEF
4 GHI	5 JKL	6 MNO
7 PQRS	8 TUV	9 WXYZ
	0	⌫

Figure 58: Customer app Birth date page of Foodpatrons

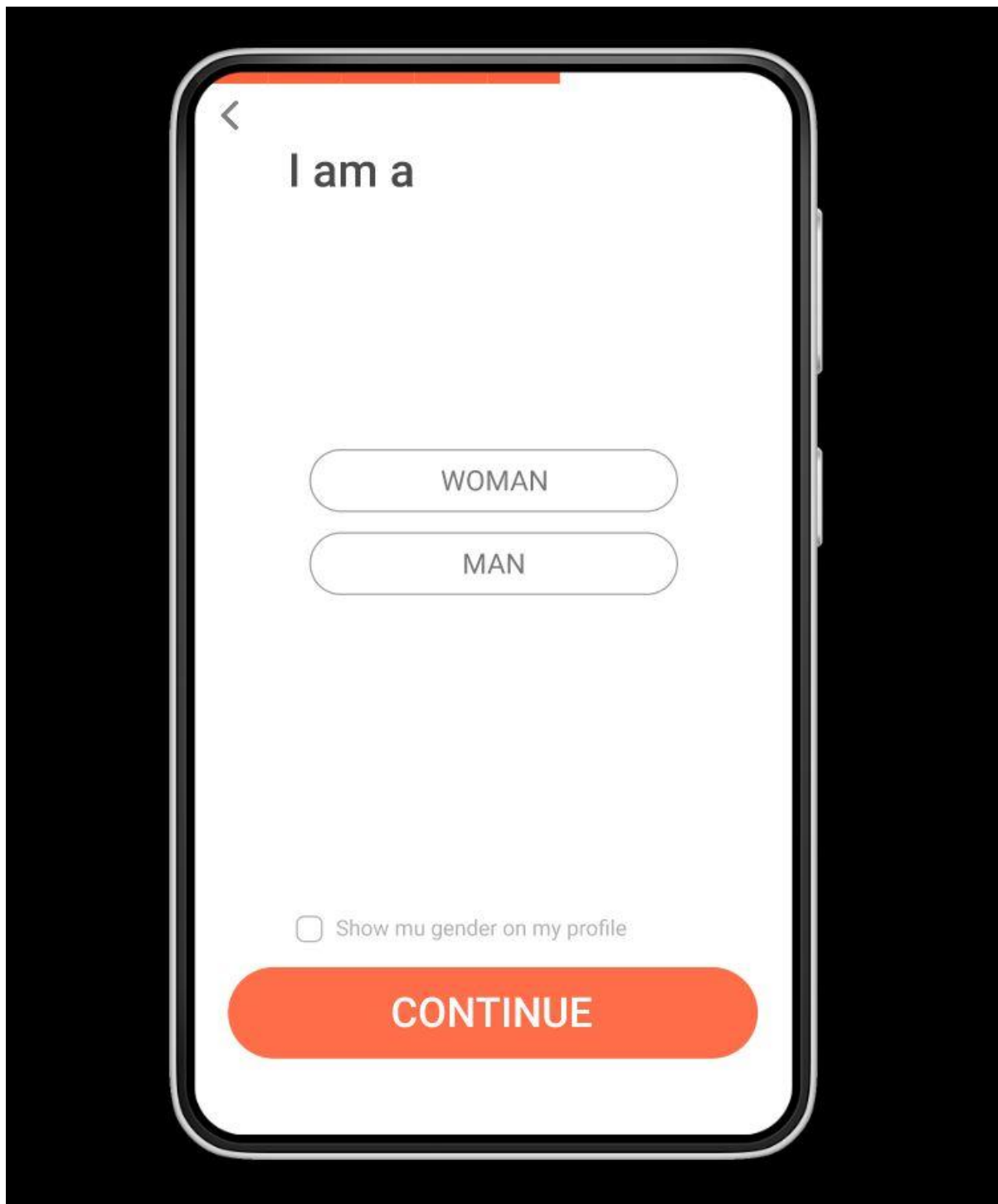


Figure 59: Customer app Gender selection page of Foodpatrons

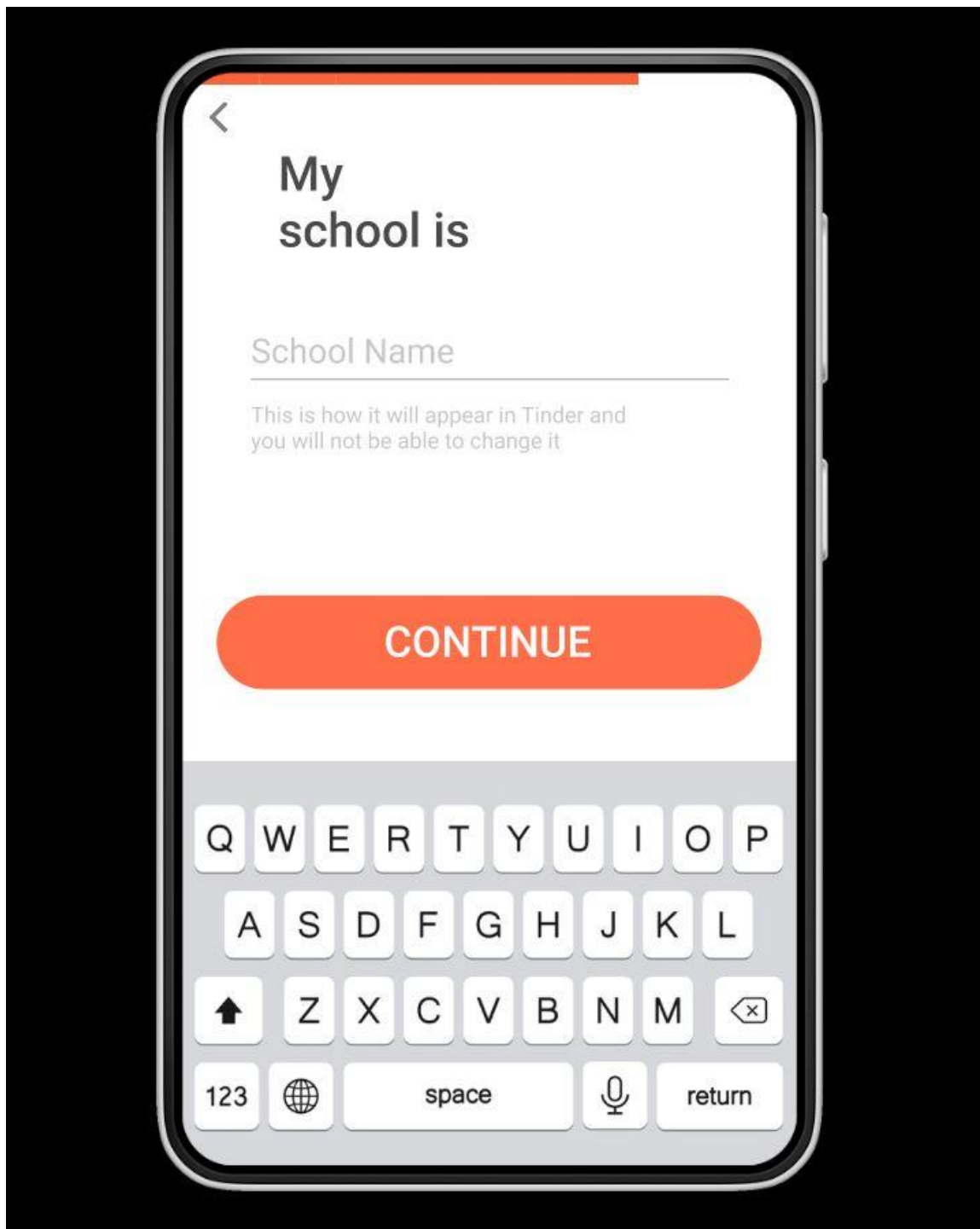


Figure 60: Customer app School name page of Foodpatrons

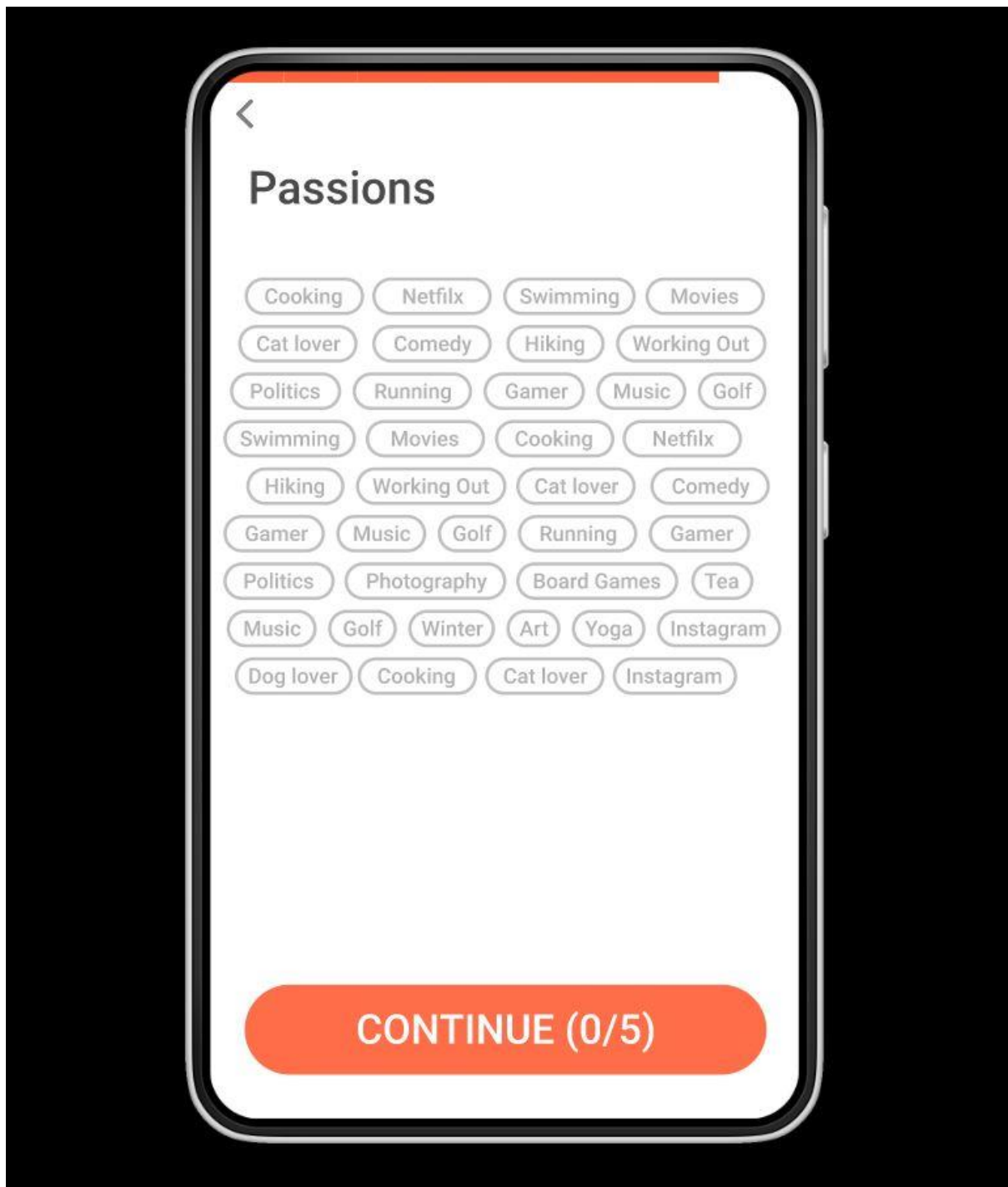


Figure 61: Customer app Passion's page of Foodpatrons

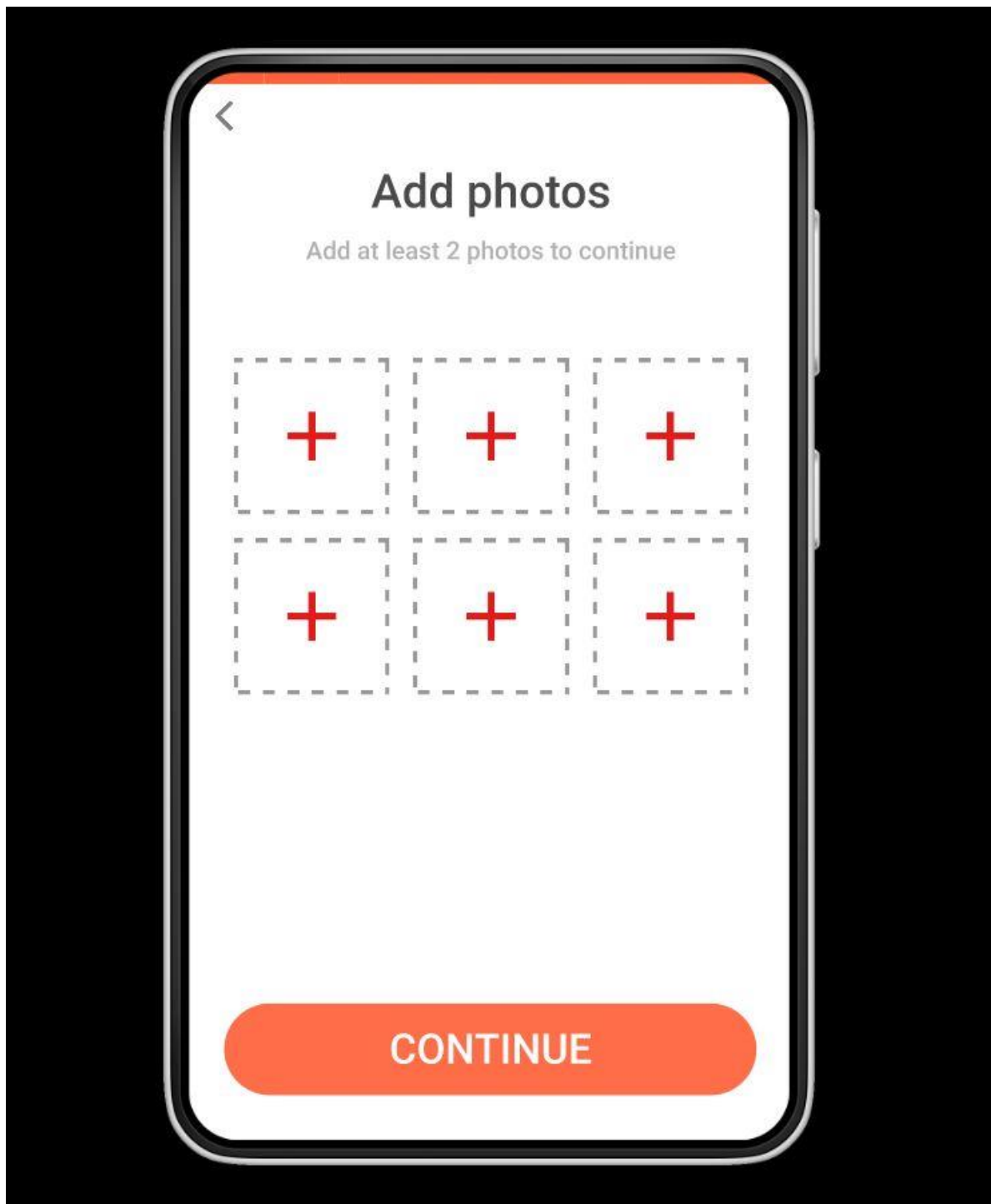


Figure 62: Customer app Add photos page of Foodpatrons

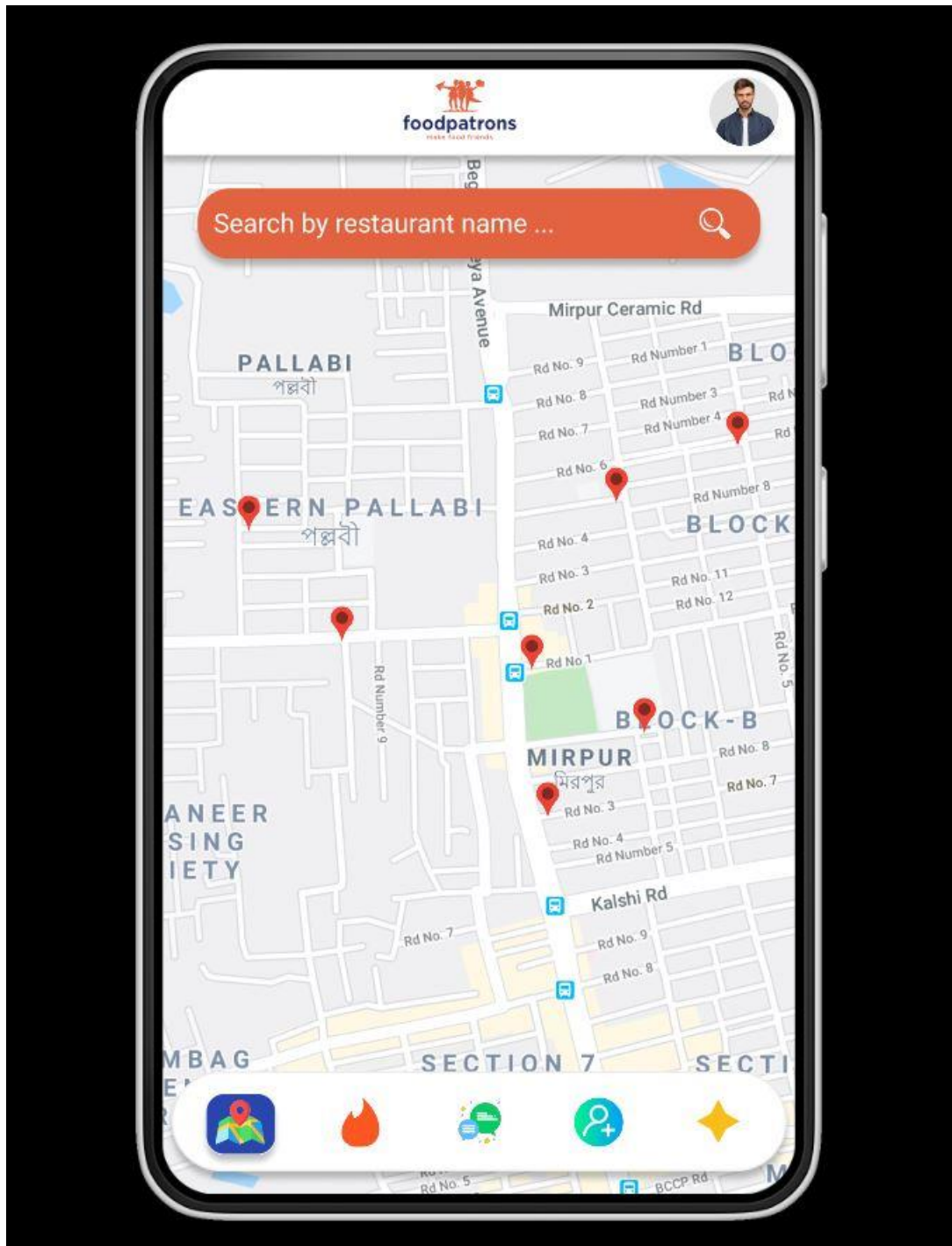


Figure 63: Customer app home page of Foodpatrons

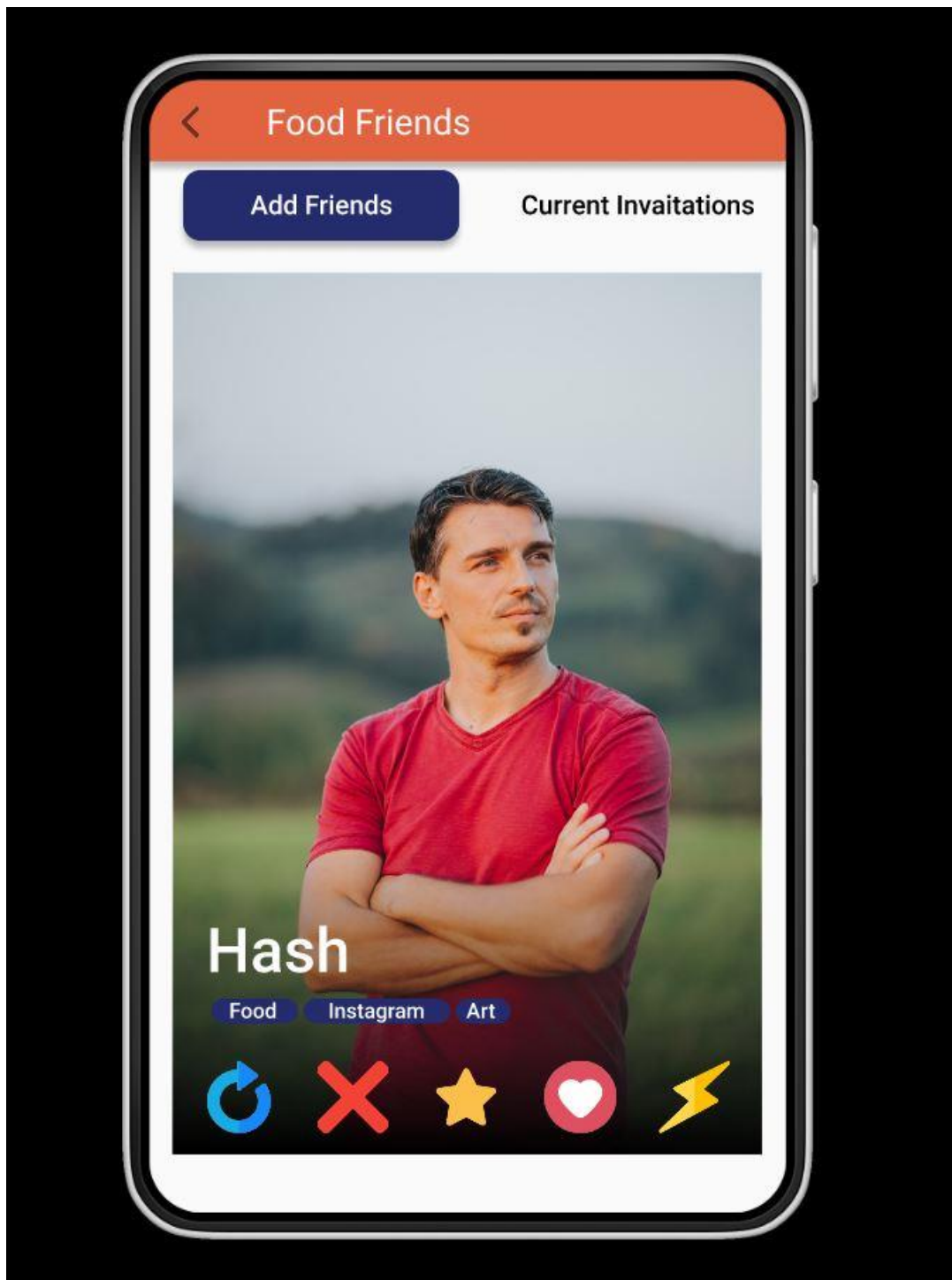


Figure 64: Customer app Partner selection page of Foodpatrons

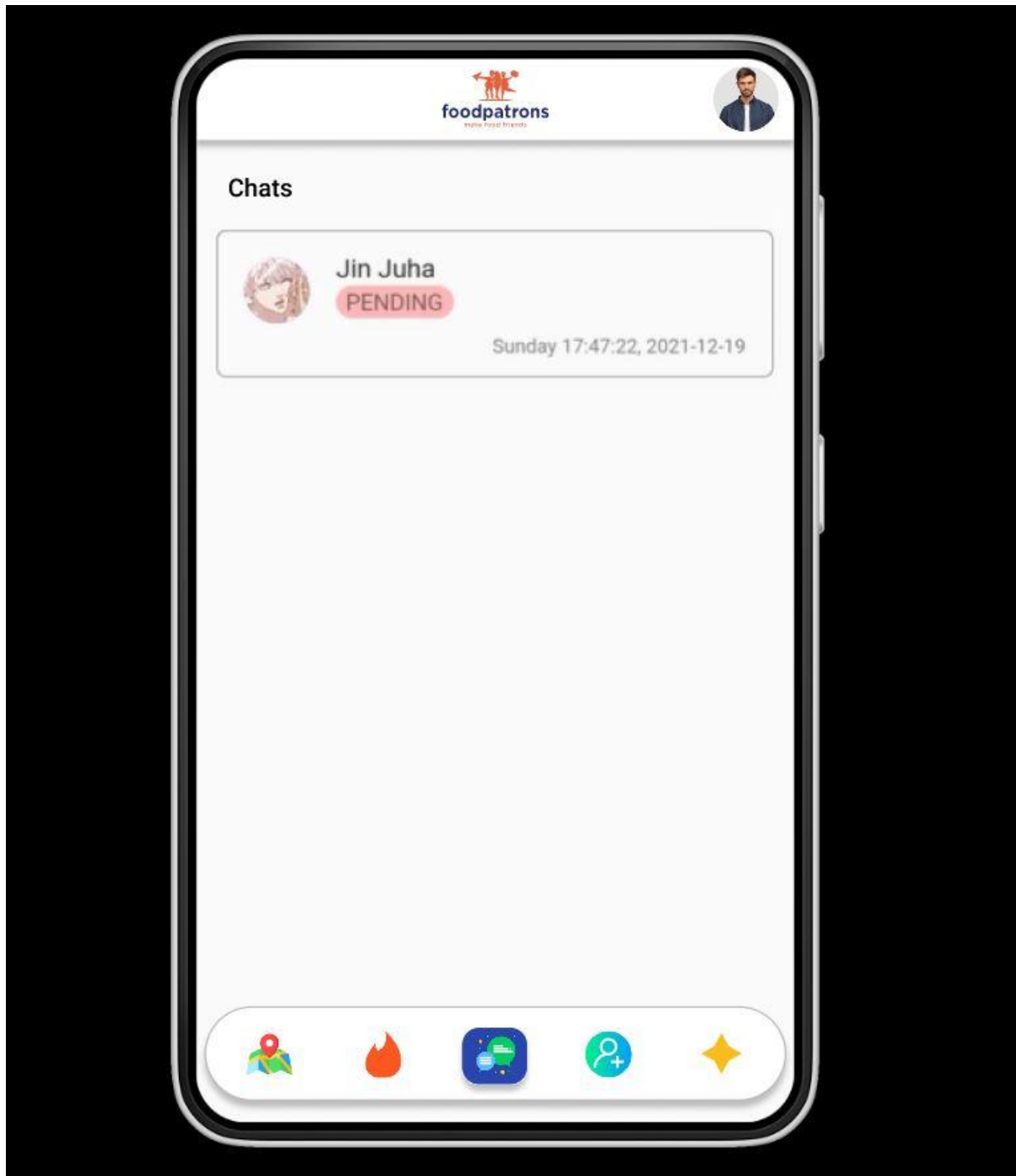


Figure 65: Customer app Chat box page of Foodpatrons

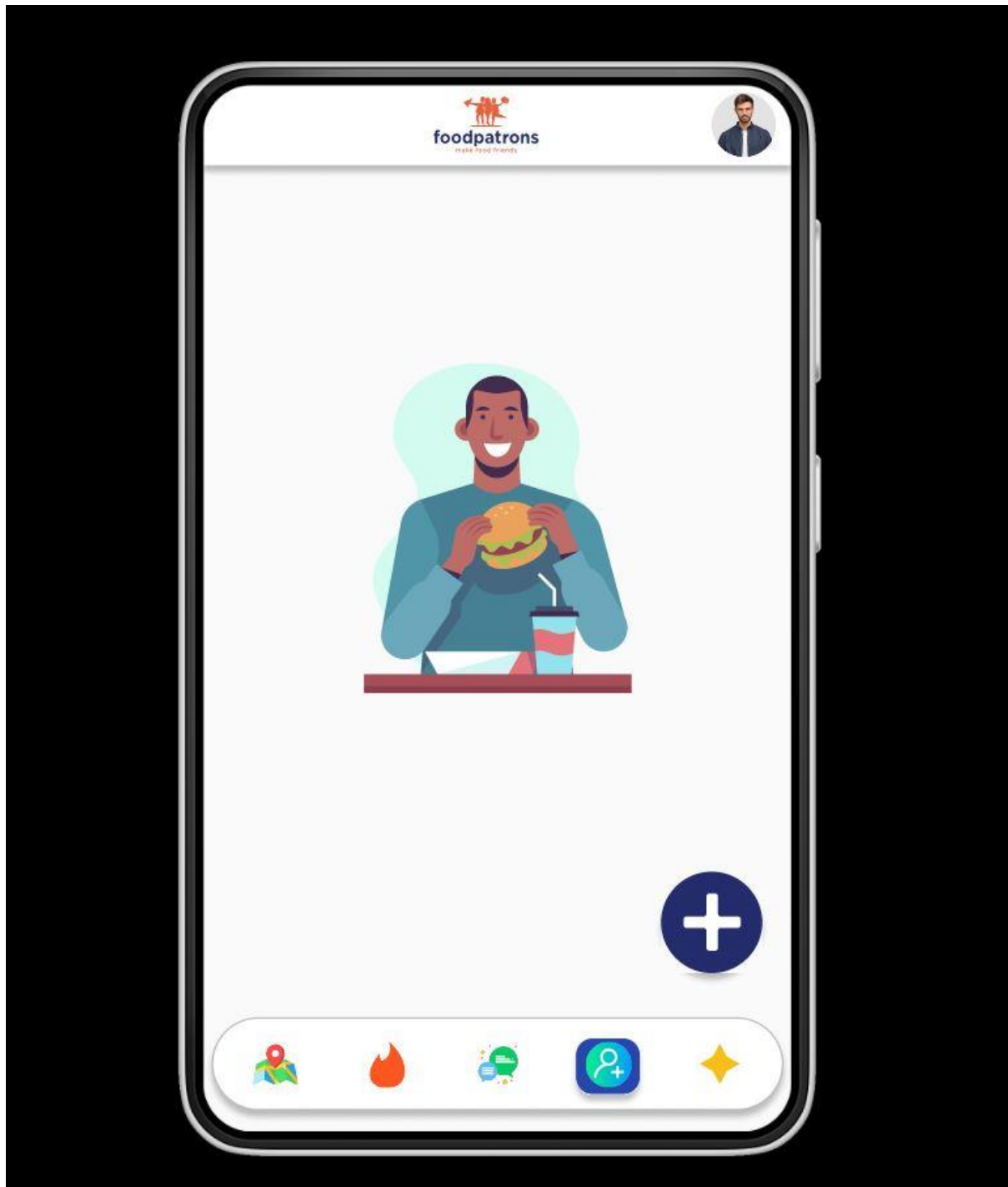
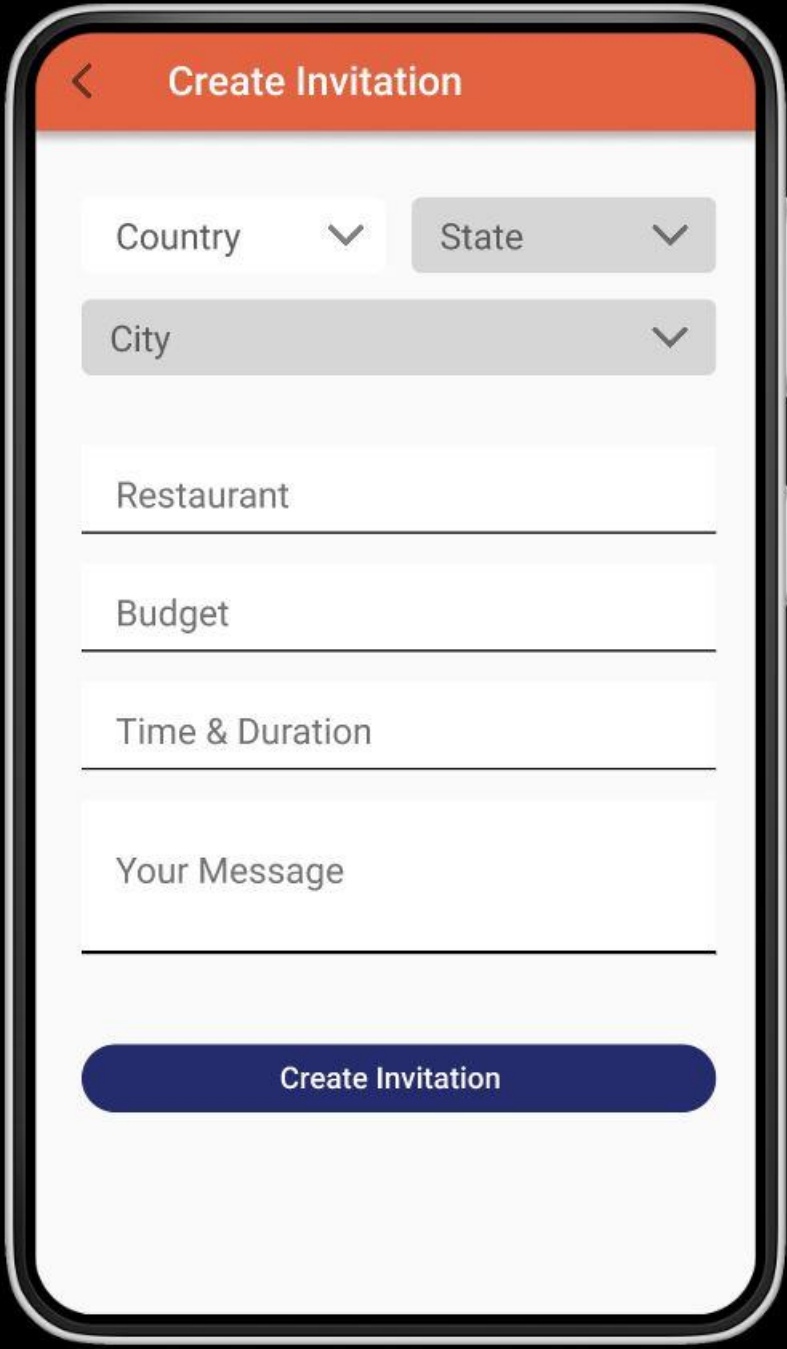


Figure 66: Customer app Add booking page of Foodpatrons



The image shows a mobile application interface for creating an invitation. The title bar at the top is orange with a back arrow and the text "Create Invitation". Below the title bar, there are three dropdown menus for location selection: "Country", "State", and "City". Each dropdown has a downward arrow icon. Below these are four text input fields: "Restaurant", "Budget", "Time & Duration", and "Your Message". Each input field has a horizontal line below it. At the bottom of the form is a dark blue button with the text "Create Invitation".

Figure 67: Customer app Add booking form page of Foodpatrons

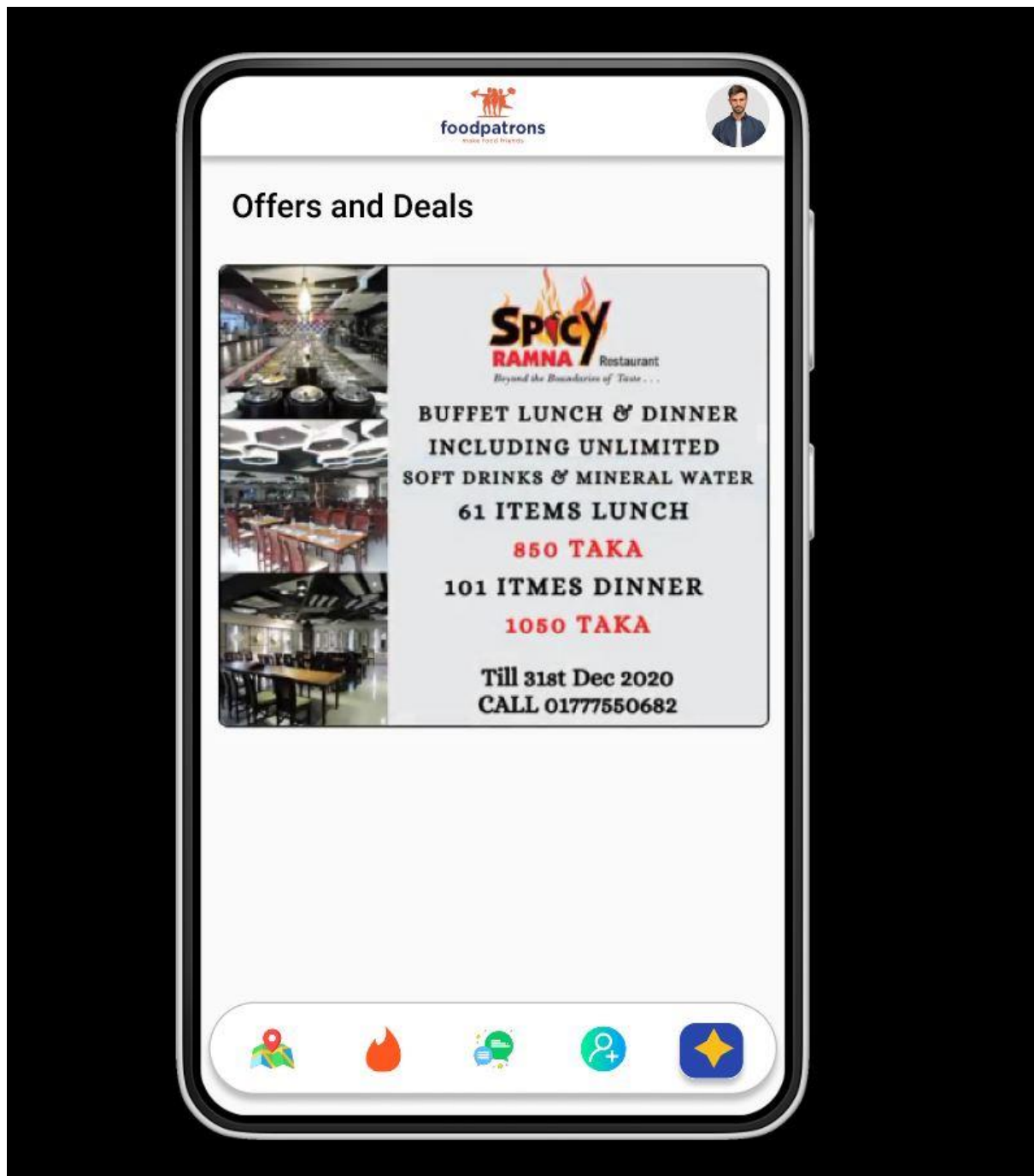


Figure 68: Customer app Restaurants offers page of Foodpatrons

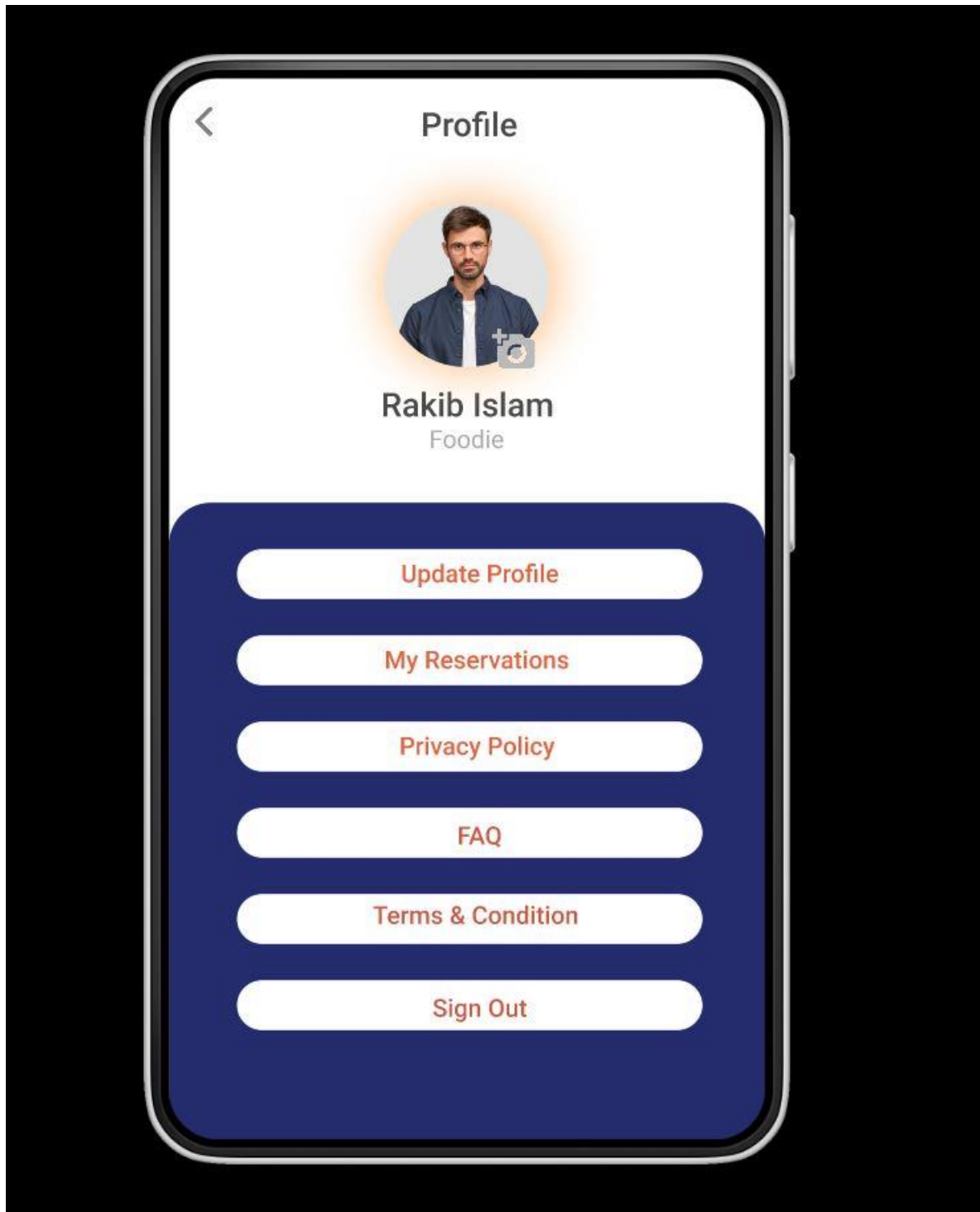


Figure 69: Customer app profile page of Foodpatrons

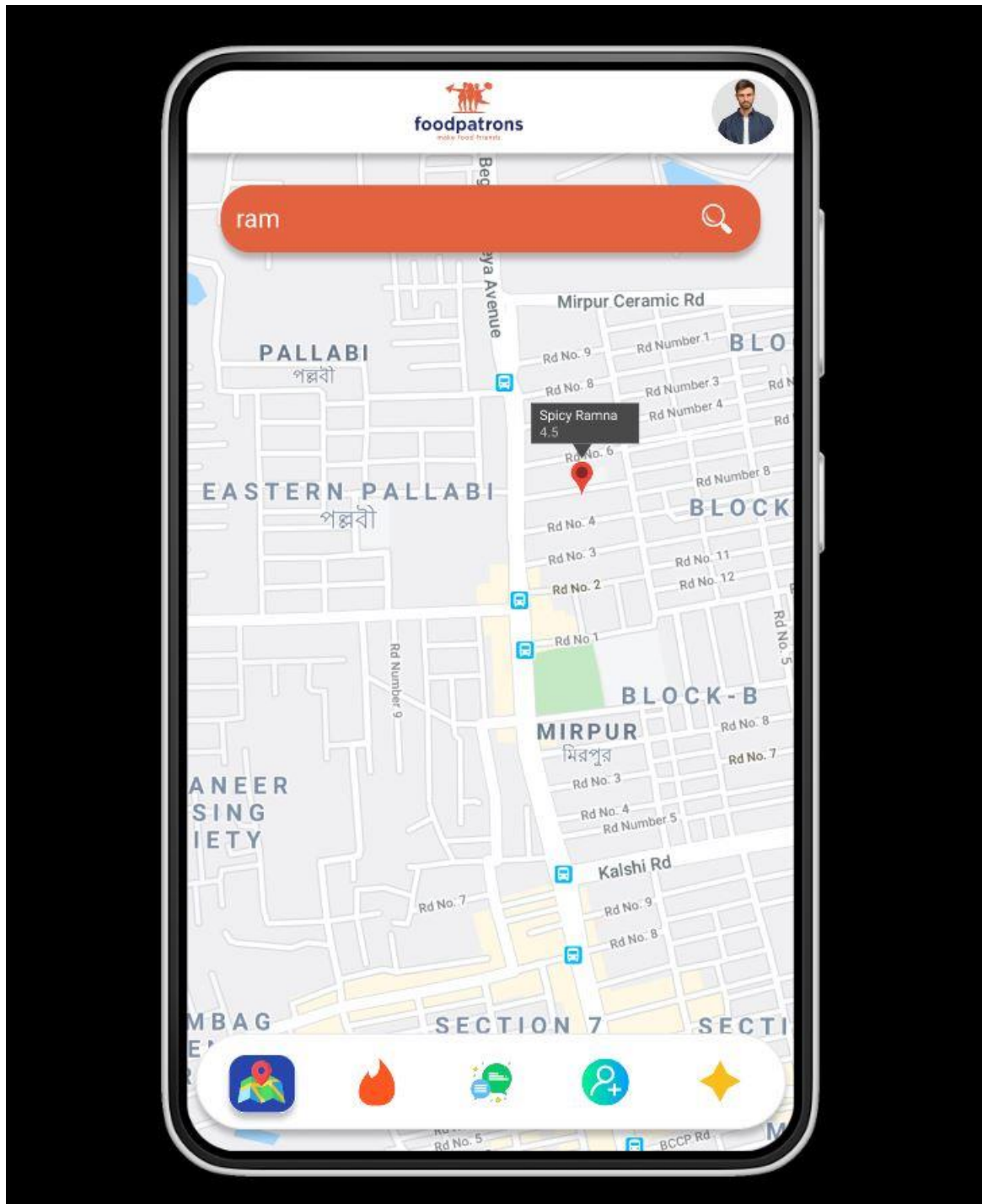


Figure 70: Customer app Restaurants Search page of Foodpatrons

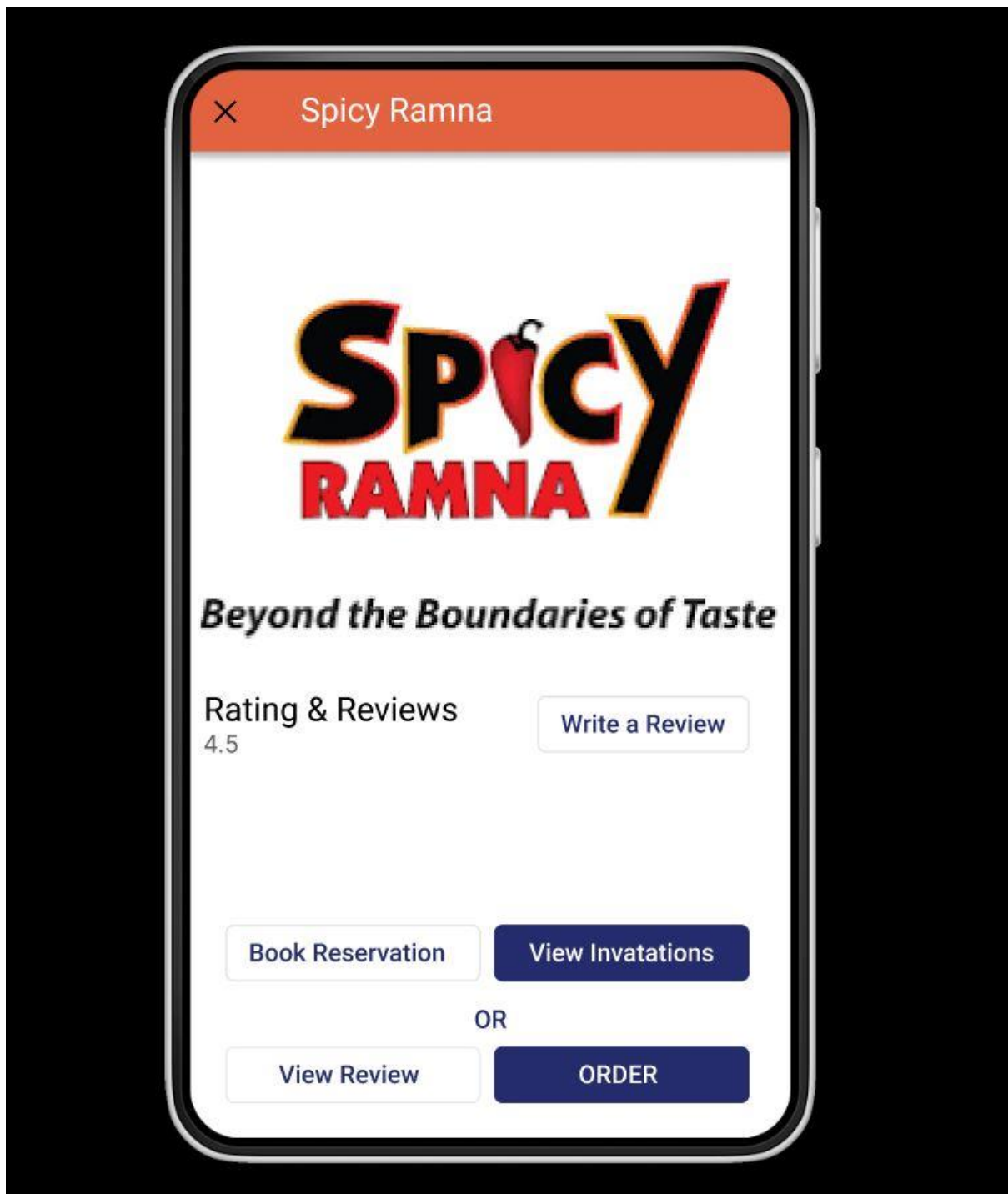


Figure 80: Customer app Restaurant's page of Foodpatrons

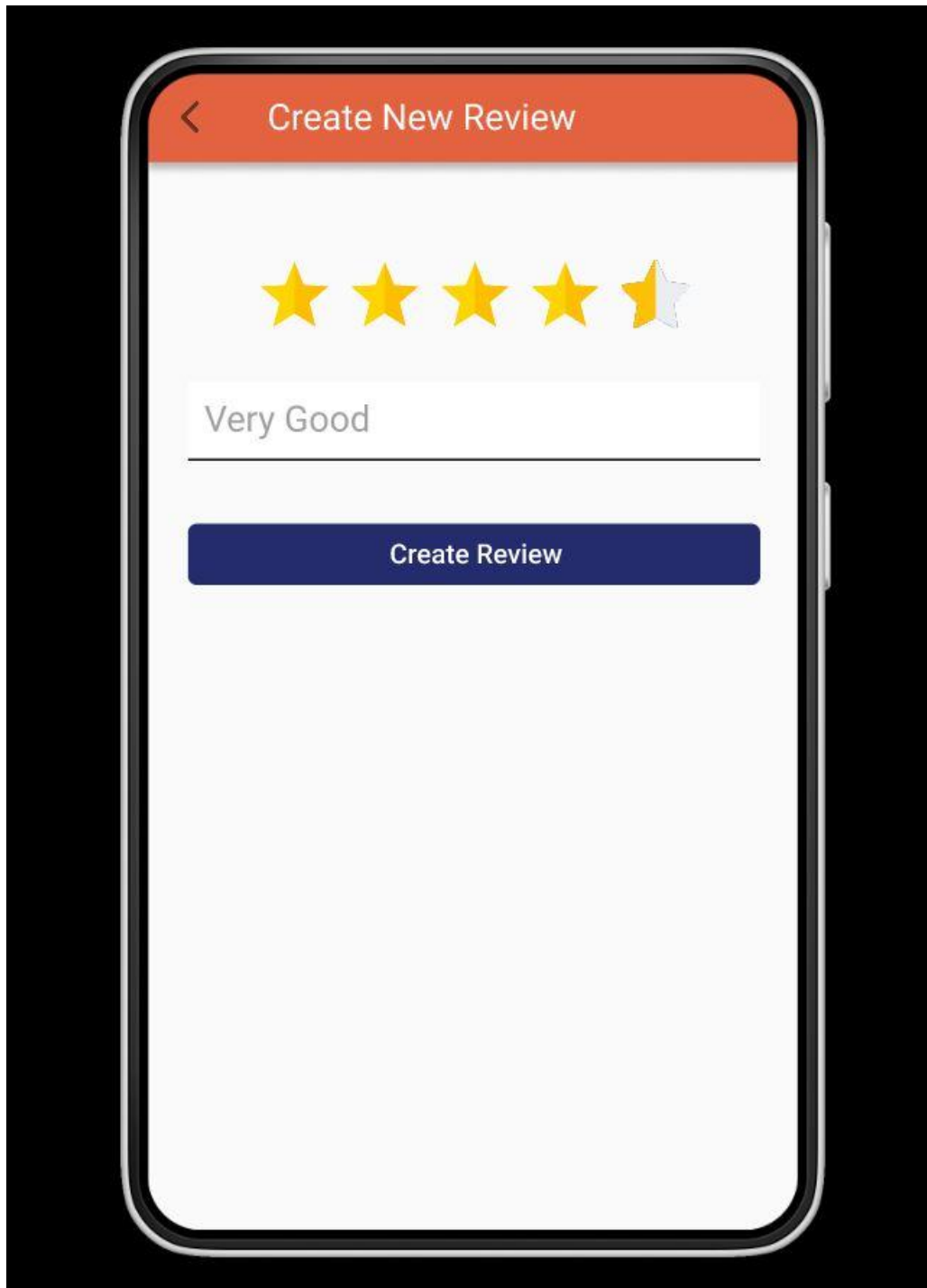


Figure 81: Customer app Restaurant's create review page of Foodpatrons

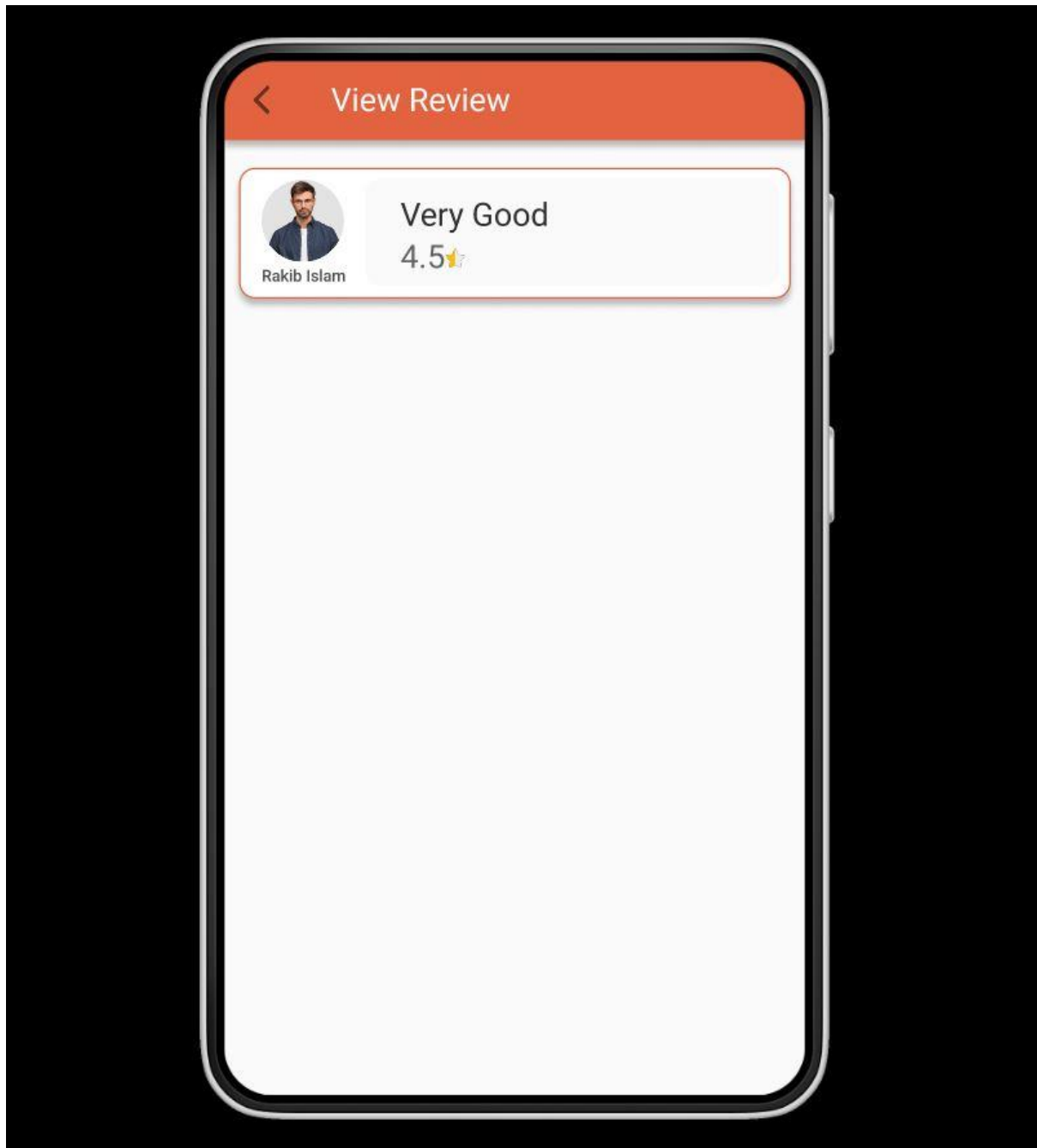


Figure 82: Customer app Restaurant's view review page of Foodpatrons

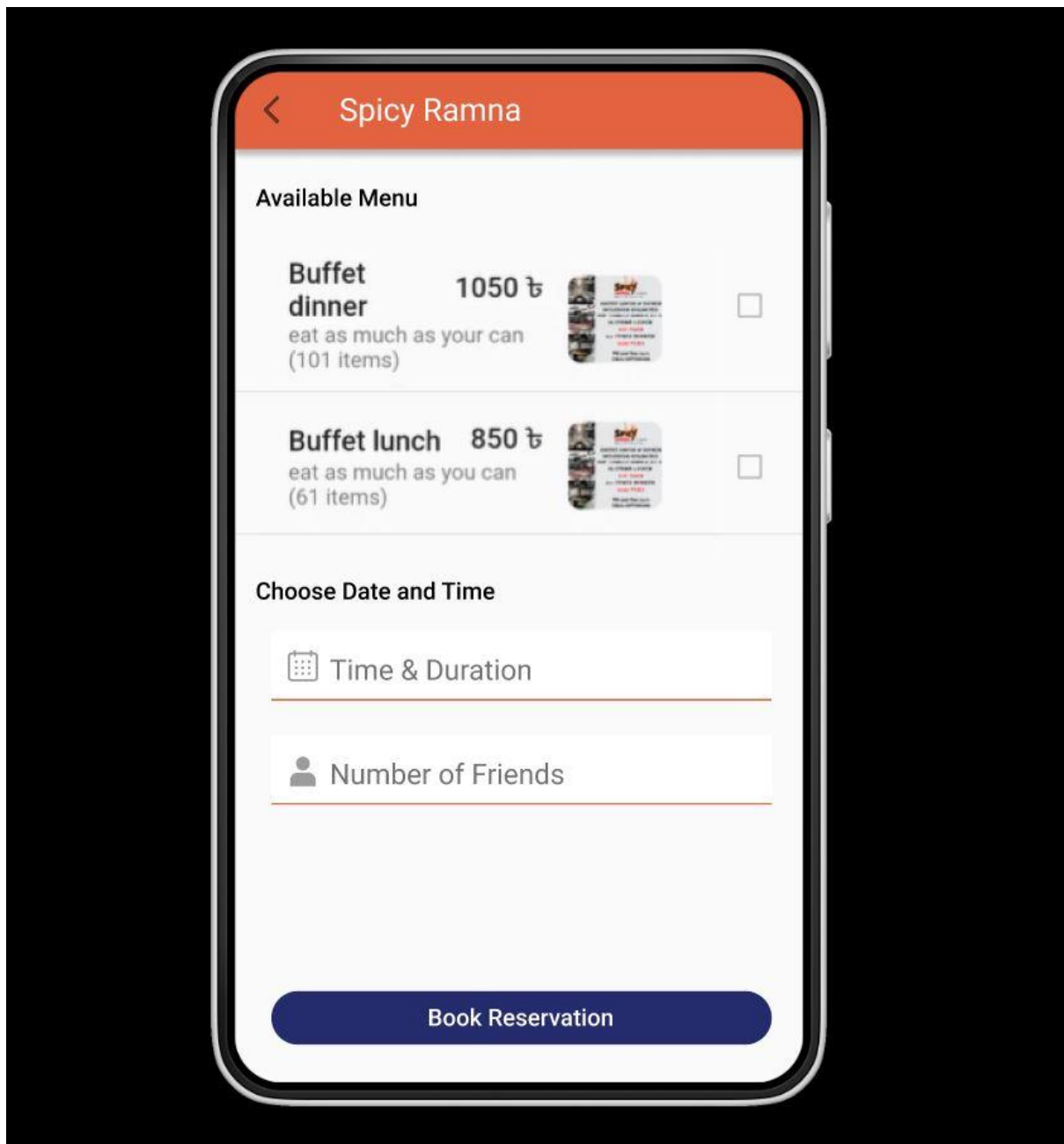


Figure 83: Customer app Restaurant's booking date form page of Foodpatrons

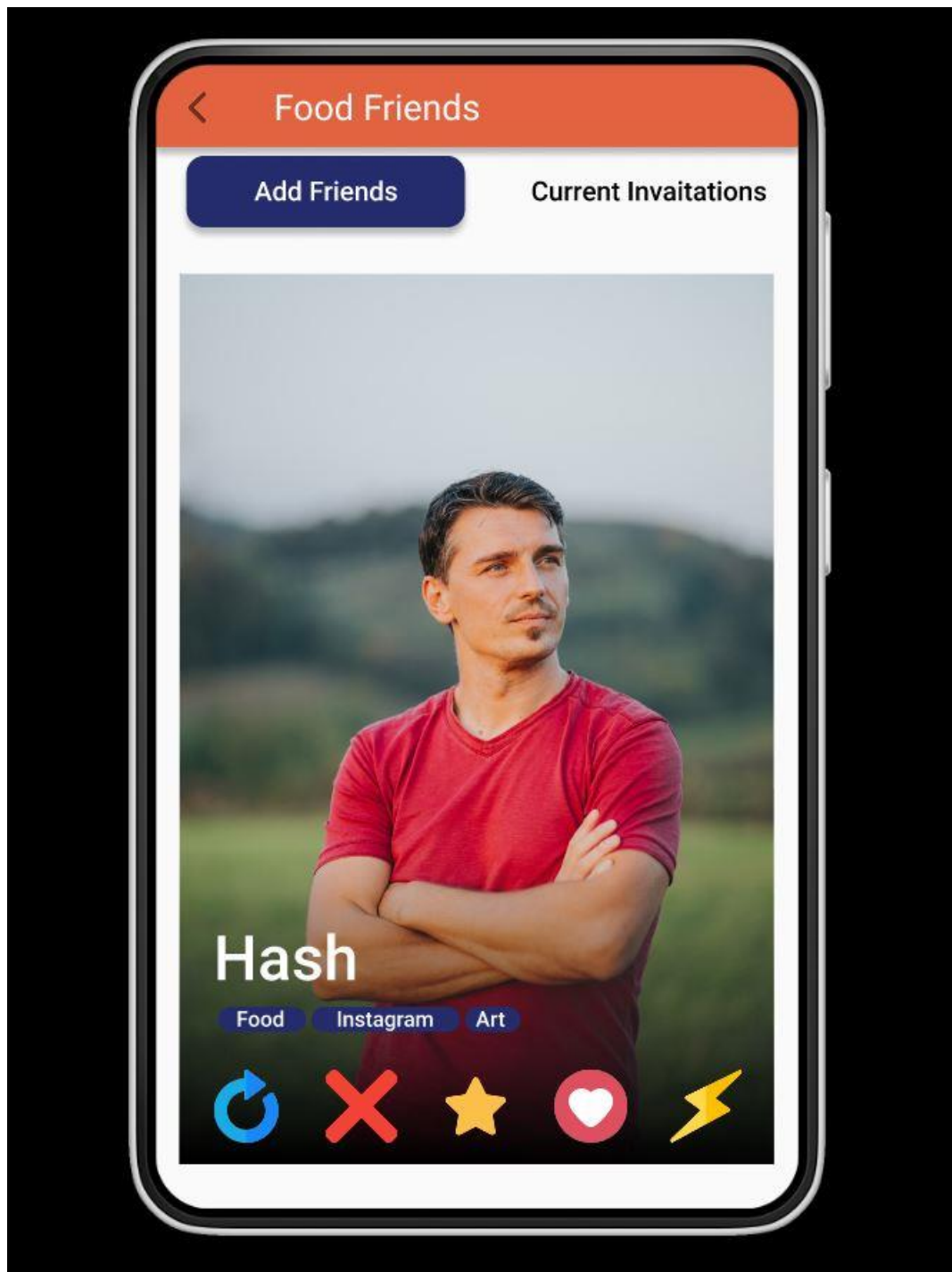


Figure 84: Customer app sent invitation page of Foodpatrons

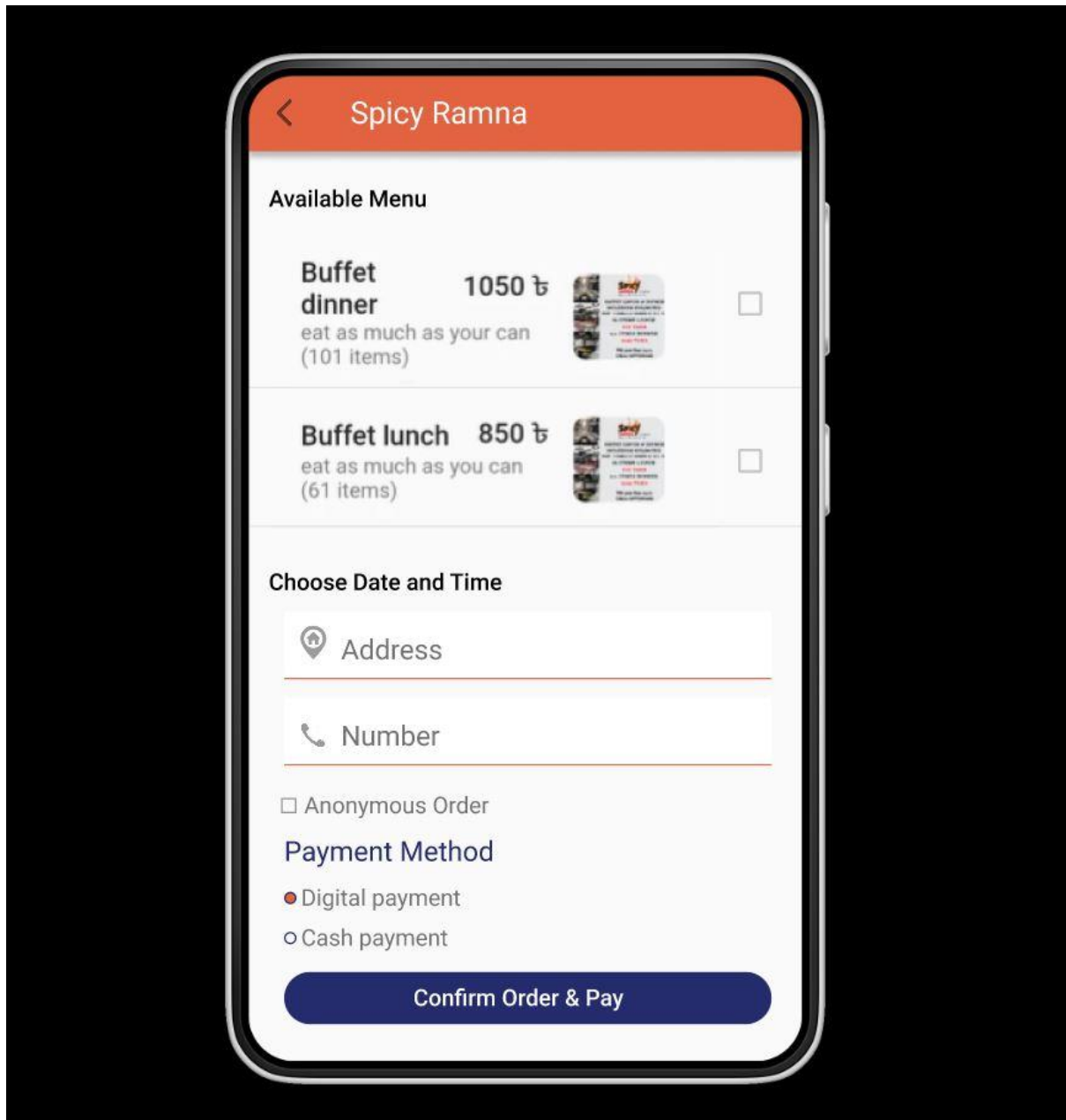


Figure 85: Customer app Food order form page of Foodpatrons

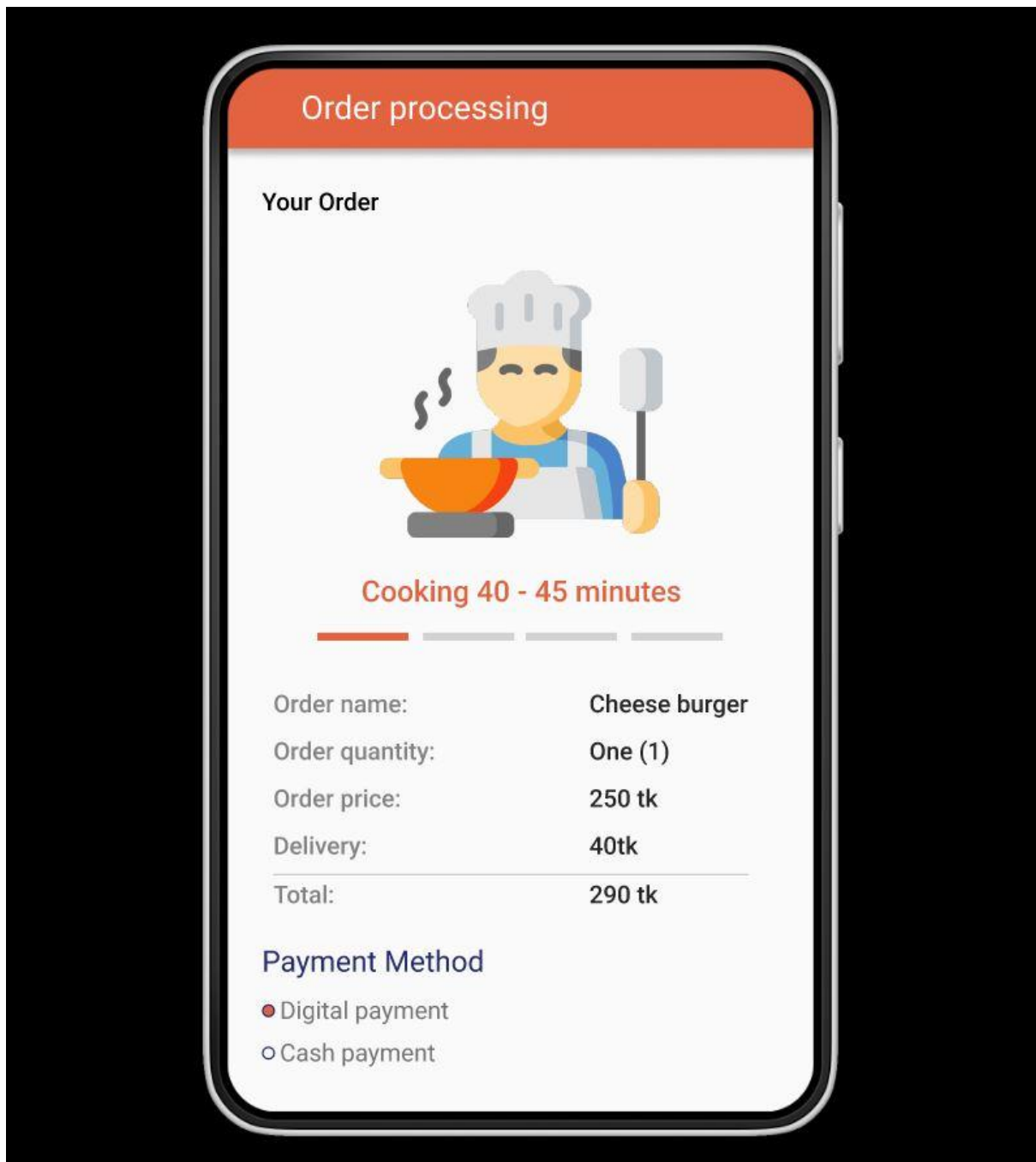


Figure 86: Customer app Food order Processing page (1) of Foodpatrons

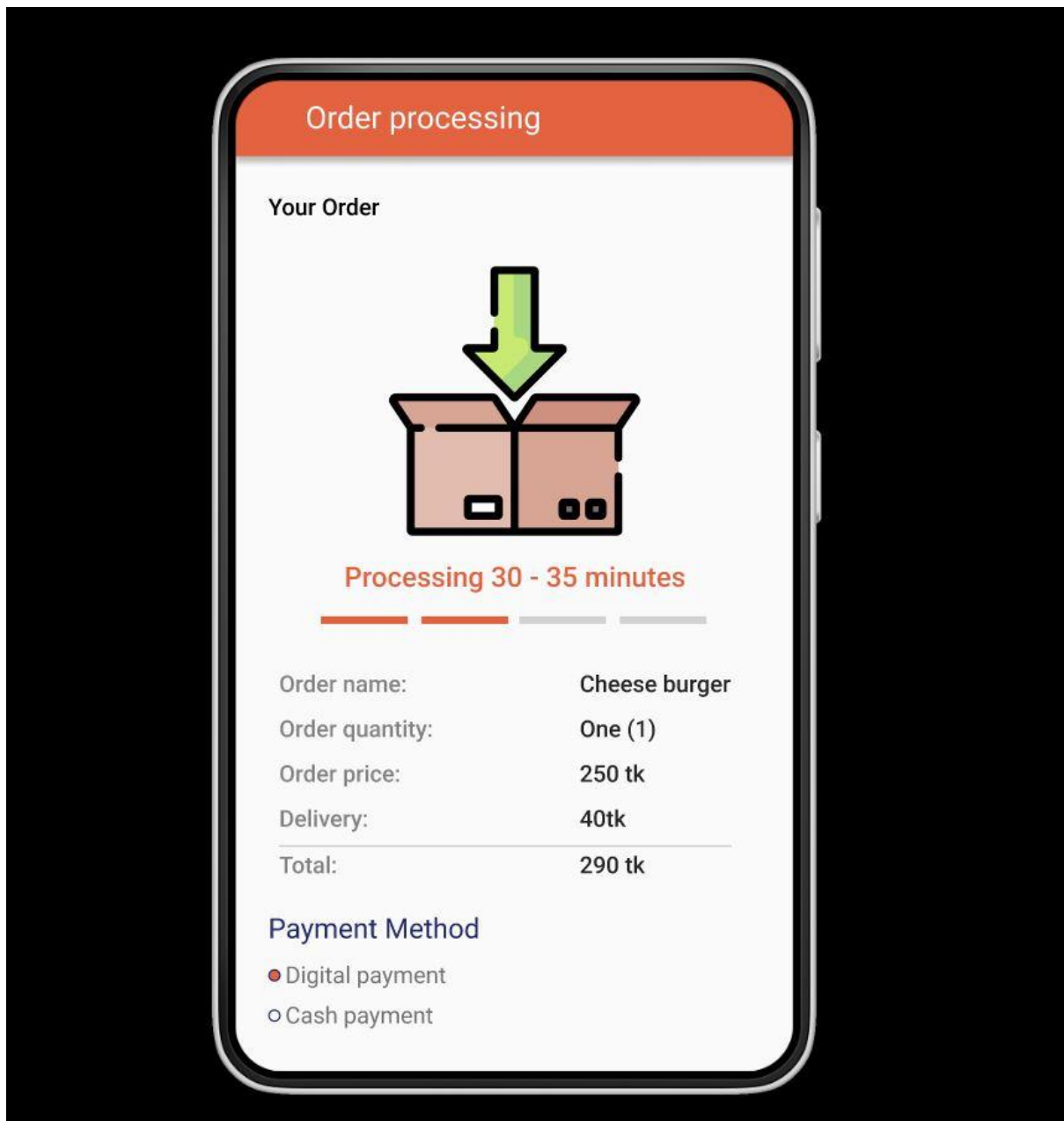


Figure 87: Customer app Food order Processing page (2) of Foodpatrons

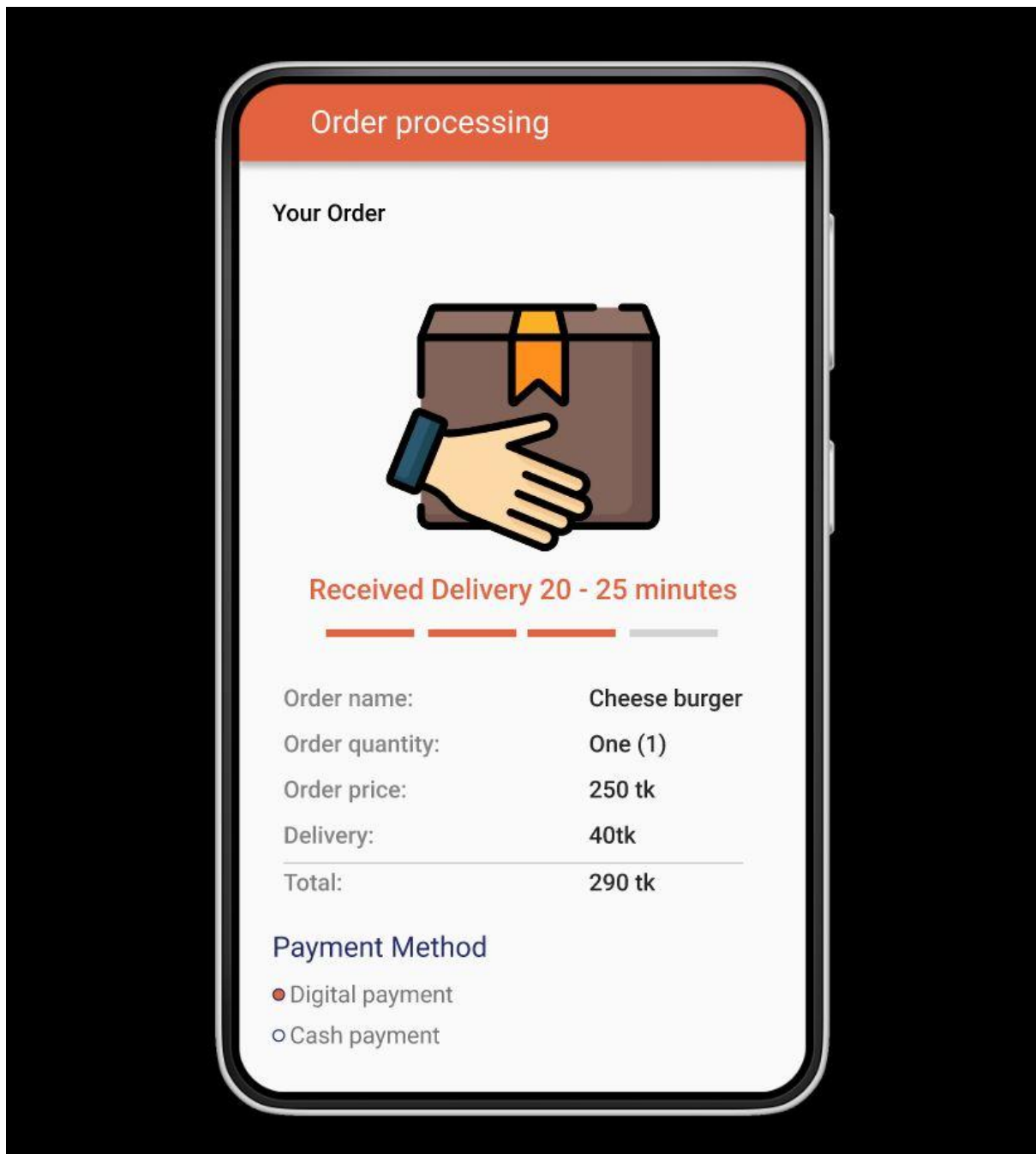


Figure 88: Customer app Food order Processing page (3) of Foodpatrons

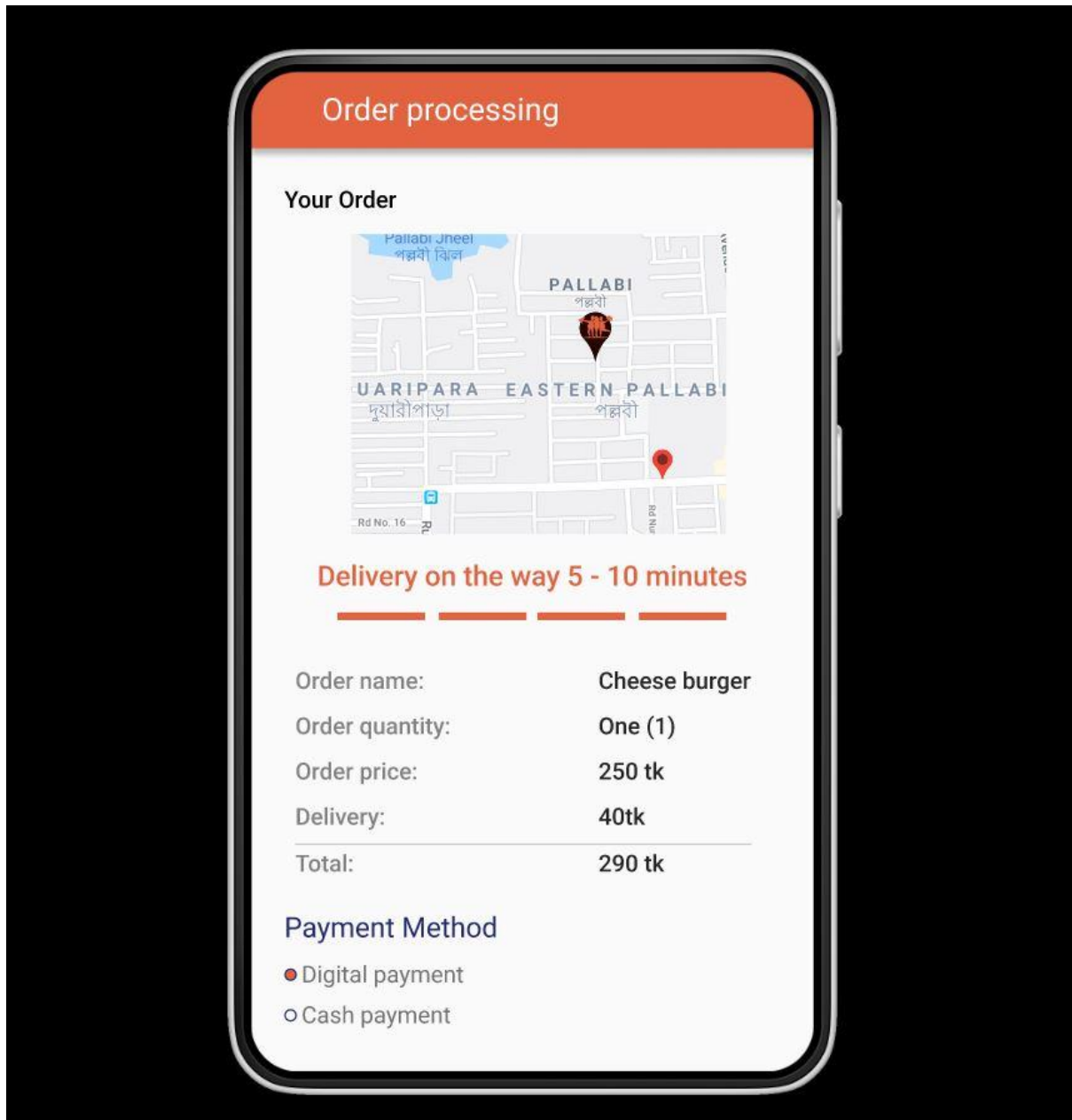


Figure 89: Customer app order current location tracking page of Foodpatrons

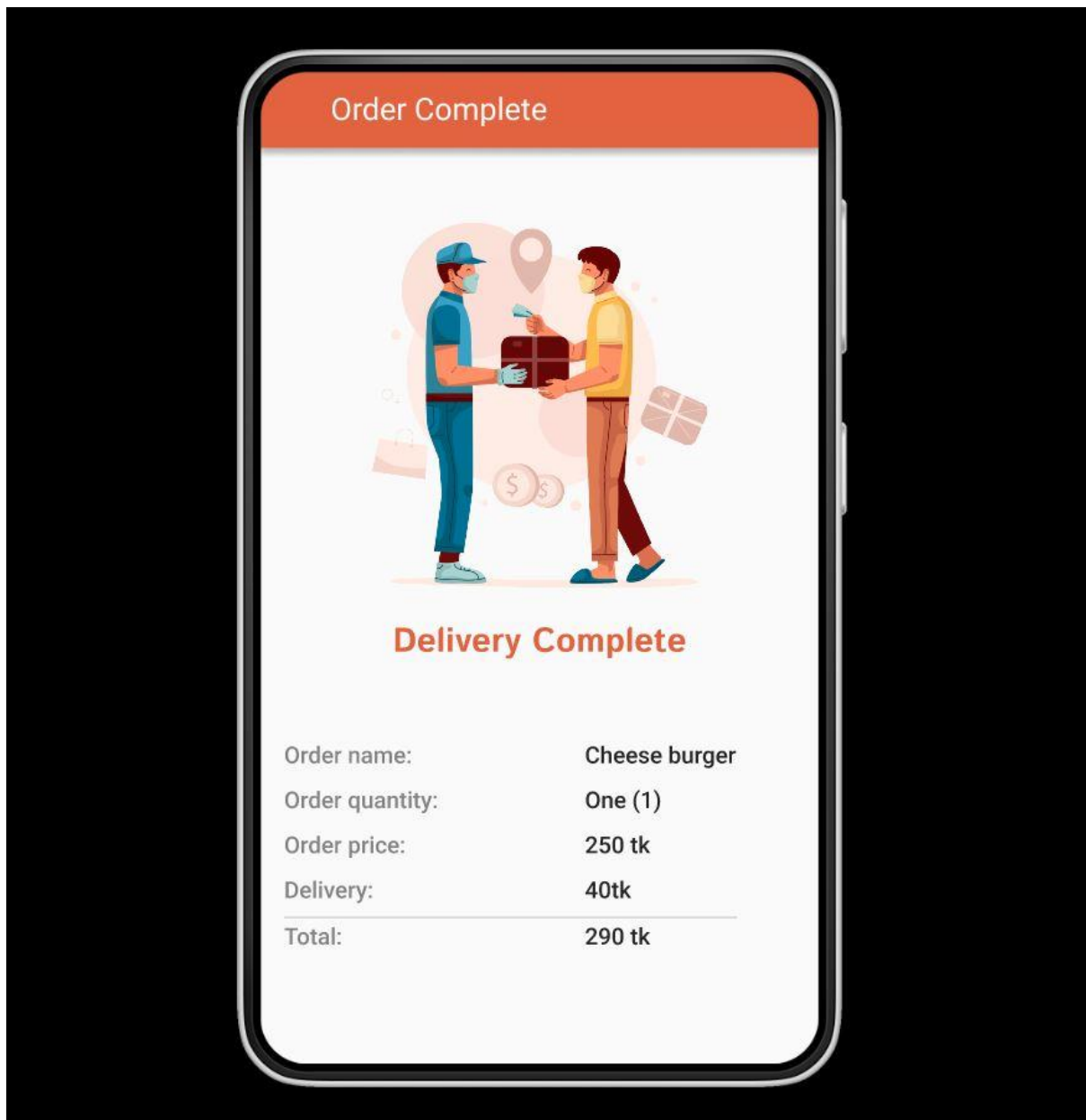


Figure 90: Customer app delivery complete page of Foodpatrons

Chapter 10 – Deployment / Development

10.1 Core Module Coding Samples

```
class PhotoPickerScreen extends ConsumerWidget {
  const PhotoPickerScreen({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context, WidgetRef ref) {
    var _provider = ref.watch(authChangeNotifierProvider);

    return SingleChildScrollView(
      padding: const EdgeInsets.symmetric(horizontal: 30.0),
      child: Column(
        children: [
          Text(
            "Add Photos",
            style: Theme.of(context).textTheme.headline5,
            textAlign: TextAlign.start,
          ),
          SizedBox(height: 6),
          Text("Add at least 2 photos to continue"),
          SizedBox(height: 100),
          GridView.count(
            crossAxisCount: 3,
            crossAxisSpacing: 8.0,
            mainAxisSpacing: 8.0,
            shrinkWrap: true,
            physics: ClampingScrollPhysics(),
            children: List.generate(
              _provider.multipleImages.length,
              (index) => GestureDetector(
                onTap: () => _pickImageToIndex(index, context, _provider),
                child: DottedBorder(
                  borderType: BorderType.RRect,
                  radius: Radius.circular(4),
                  strokeWidth: 3,
                  dashPattern: [8, 4],
                ),
              ),
            ),
          ),
        ],
      ),
    );
  }
}
```

Ln 1, Col 1 - Spaces: 2 - 1

Figure 91: Add photos

```

class CreateNewReview extends StatefulWidget {
  final int? restaurantId;
  final int? customerId;

  const CreateNewReview({Key? key, this.restaurantId, this.customerId})
    : super(key: key);

  @override
  _CreateNewReviewState createState() => _CreateNewReviewState();
}

class _CreateNewReviewState extends State<CreateNewReview> {
  double _rating = 4.0;
  String? _reviewMessage;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: AppBarTitle(
          title: 'Create New Review',
        ),
      ),
      body: Form(
        child: ListView(
          padding: const EdgeInsets.all(16),
          children: [
            Center(
              child: RatingBar.builder(
                initialRating: 4,
                minRating: 1,
                direction: Axis.horizontal,
                allowHalfRating: true,
                itemCount: 5
              ),
            ),
          ],
        ),
      ),
    );
  }
}

```

Figure 92: Create review

```

class PhoneNumberScreen extends ConsumerStatefulWidget {
  const PhoneNumberScreen({Key? key}) : super(key: key);

  @override
  _PhoneNumberScreenState createState() => _PhoneNumberScreenState();
}

class _PhoneNumberScreenState extends ConsumerState<PhoneNumberScreen> {
  final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
  PhoneNumber _number = PhoneNumber(isoCode: 'BD');
  bool _isValidNumber = false;
  bool _isLoading = false;

  void _setIsValidNumber(bool validity) {
    setState(() {
      _isValidNumber = validity;
    });
  }

  @override
  void initState() {
    super.initState();
  }

  @override
  Widget build(BuildContext context) {
    final _provider = ref.read(authChangeNotifierProvider);
    return Scaffold(
      appBar: AppBar(
        elevation: 0.0,
        backgroundColor: Colors.transparent,
        // systemOverlayStyle: SystemUiOverlayStyle(statusBarColor: brandColor),
      ),
      backgroundColor: Color(0xFFe26240),
    );
  }
}

```

Ln 1, Col 1 Space

Figure 93: Add mobile number

```

class ProfileProvider extends ChangeNotifier {
  ProfileProvider() {
    _countries = Countries.fromJson(availableCountries).countries;
    UserProfile? _profile = DbService.getUserProfile();
    if (_profile != null) {
      setUserProfileInfo(_profile);
    }
  }

  String? _gender;
  String? get gender => _gender;

  Country? _country;
  Country? get country => _country;

  CountryState? _state;
  CountryState? get state => _state;

  City? _city;
  City? get city => _city;

  List<FoodTag>? _tags = [];
  List<FoodTag>? get tags => _tags;

  String? _interestedIn;
  String? get interestedIn => _interestedIn;

  List<Country>? _countries = [];
  List<Country>? get countries => _countries;

  List<CountryState> _states = <CountryState>[];
  List<CountryState> get states => _states;

  List<City> _cities = <City>[];

```

Ln 1, Col 1 - Spaces: 2 - UTF-8 -

Figure 94: profile change

```

const UploadImageScreen({Key? key, this.imagePath}) : super(key: key);

@override
UploadImageScreenState createState() => _UploadImageScreenState();
}

class _UploadImageScreenState extends ConsumerState<UploadImageScreen> {
  bool _isLoading = false;

  @override
  Widget build(BuildContext context) {
    File _image = File(widget.imagePath!);
    return Scaffold(
      appBar: AppBar(
        title: AppBarTitle(title: 'Upload Profile Picture'),
      ),
      bottomSheet: Padding(
        padding: const EdgeInsets.symmetric(horizontal: 16.0, vertical: 10),
        child: _isLoading
          ? Container(
              height: 60,
              child: Center(child: CircularProgressIndicator()),
            )
          : SizedBox(
              width: double.infinity,
              child: ElevatedButton(
                style: ElevatedButton.styleFrom(
                  shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(16),
                  ),
                ),
                onPressed: () async {
                  setState(() {

```

Figure 95: Update profile

10.2 Possible problem breaks down

The entire project must be divided down into discrete jobs to make system development easier and more effective. The following are some possible breakdowns for the proposed project:

- Analysis and database design
- Admin panel management development
- Customer panel management
- Customer panel development
- Voluntary organization panel development

Analysis and Database Design

- Determine and complete the requirements
- Gather information in order to cope with the situation
- Organize the data you've acquired.
- Create an ERD and a data dictionary.
- Create the schema using the data dictionary as a guide.

Admin panel development

- create the necessary pages
- Create a mechanism for admin manipulation.
- Create a framework for manipulating partners
- Create approval options for customers, restaurant owners, and delivery men.

Customer panel development

- Front-end design
- Registration and login system
- Order request manipulation
- Anonymous order request management

Customer panel development

- Front-end design, registration and login system.
- Location based request filtering and notification system development
- Request management

Voluntary organization panel development

- Front-end and the registration and login system

- Anonymous order request management
- Anonymous order database manipulation

10.3 Prioritization while developing

We've already prioritized the criteria list to determine the system's functionality. Now is the moment to prioritize activities during the development phase, which is critical since haphazard development may leave certain key capabilities undeveloped. The following is a list of development tasks that have been prioritized:

- Analysis & database design
- Database development
- Registration and login system implementation for the required users.
- Delivery man panel implementation
- Delivery man location tracking implementation
- Customer panel implementation.
- Voluntary organization panel implementation and
- Finally, the admin panel development.

Chapter 11 – Testing

11.1 Test Plan Acceptance

In order to create a successful software product, testing is crucial. During the analytical phase, a testing strategy should be devised. Both the user and the developer must agree to this arrangement. The project's test strategy will guarantee that the user's testing needs and acceptance criteria are met. There are two forms of testing:

➤ **Functional Testing**

Functional testing divided into three types these are-

- **Unit Testing**
 - Validation of input fields on the form
 - Customer approval filtering
 - Customer filtering from the list
 - Approved, pending and rejected Delivery man list filtering from the delivery man list.
- **Module Testing**
 - Submitting the registration and login form without data
 - Registration with the invalid data.
 - Submitting a form with proper data.
- **Integration Testing**
 - Login system with the valid credentials
 - Successful Order request placement

➤ **Non-functional Testing**

I'm going to use two different sorts of non-functional testing in this project:

- **Security Testing**
 - Role wise home page access
 - Login attempt with the invalid credential
- **Usability Testing**
 - Admin panel testing
 - Testing with the Customer

11.2 Test Case

Following the completion of the test acceptance plan, test cases must be developed. The following are some of the Foodpatrons system's test cases:

Unit test – test case:

Test Case Name	Unit Test		
Test Class			
Test Description			
Data Source	Test Steps	Expected Result	Actual Result

Module Test – test case:

Test Case Name	Module Test		
Test Class			
Test Description			
Data Source	Test Steps	Expected Result	Actual Result

Integration Testing – test case:

Test Case Name	Integration Test		
Test Class			
Test Description			
Data Source	Test Steps	Expected Result	Actual Result

--	--	--	--

11.3 Unit Testing

Unit Testing 1:

Test Case

Test Case Name	Unit Test		
Test Class	Customer Register Controller		
Test Description	Customer name validation for customer registration		
Data Source	Test Steps	Expected Result	Actual Result
	1. Filling all required fields except the customer's name field. 2. Submit the form	An error message should return that customer name must not be empty.	A message is showing that customer name is required for getting registered.

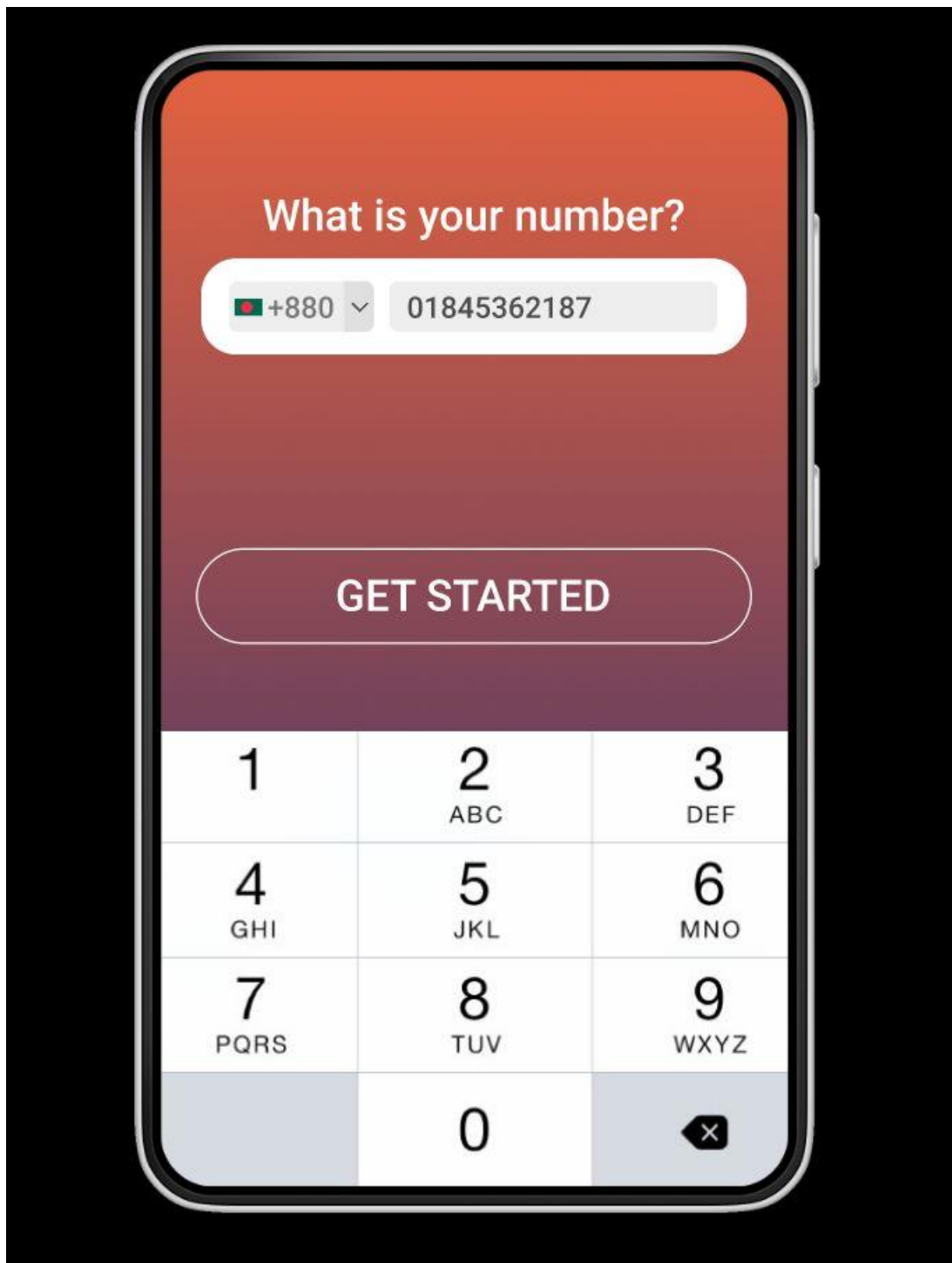


Figure 91.1: Unit Testing 1 Test Case

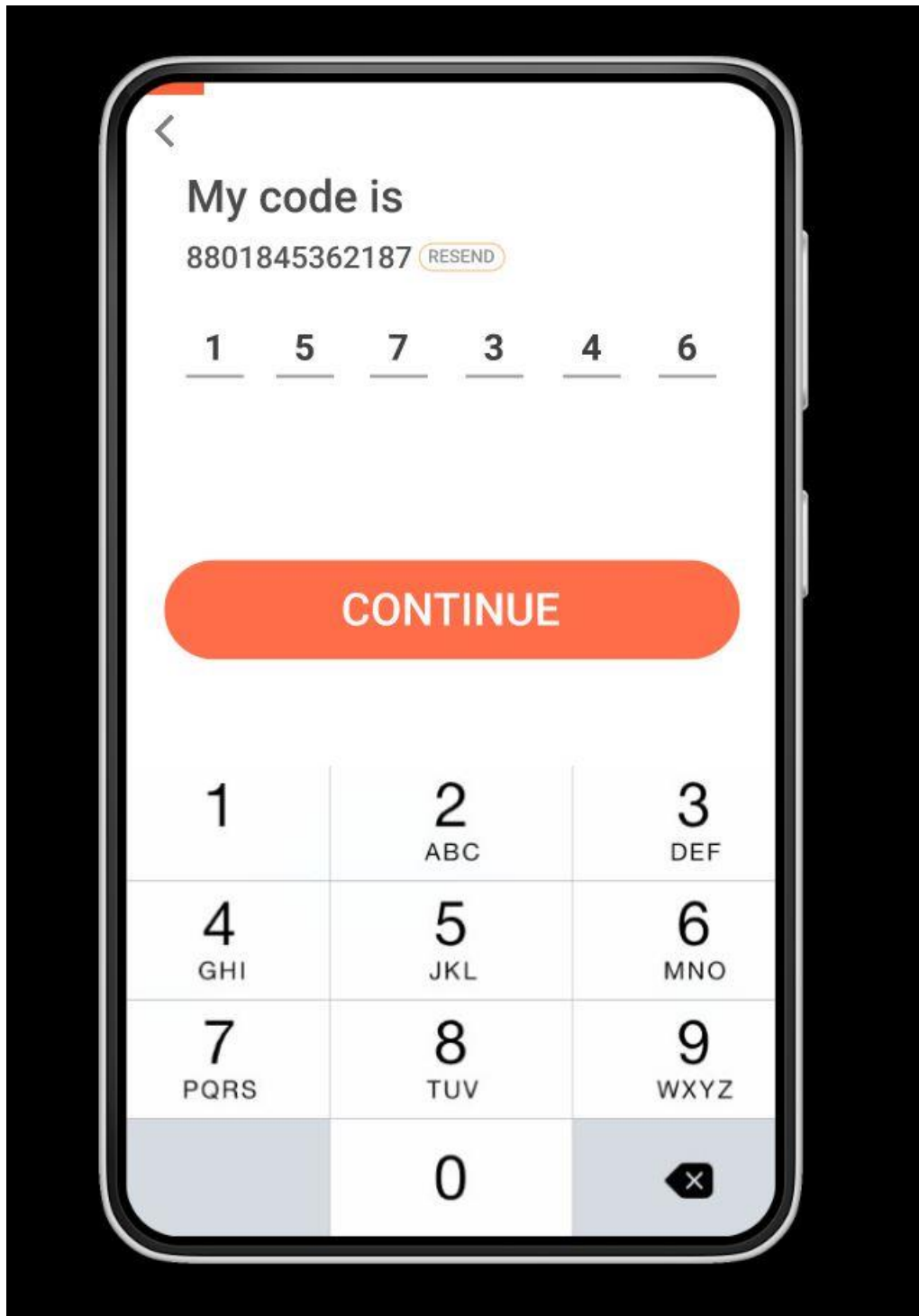


Figure 91.2: Unit Testing 1 Test Case

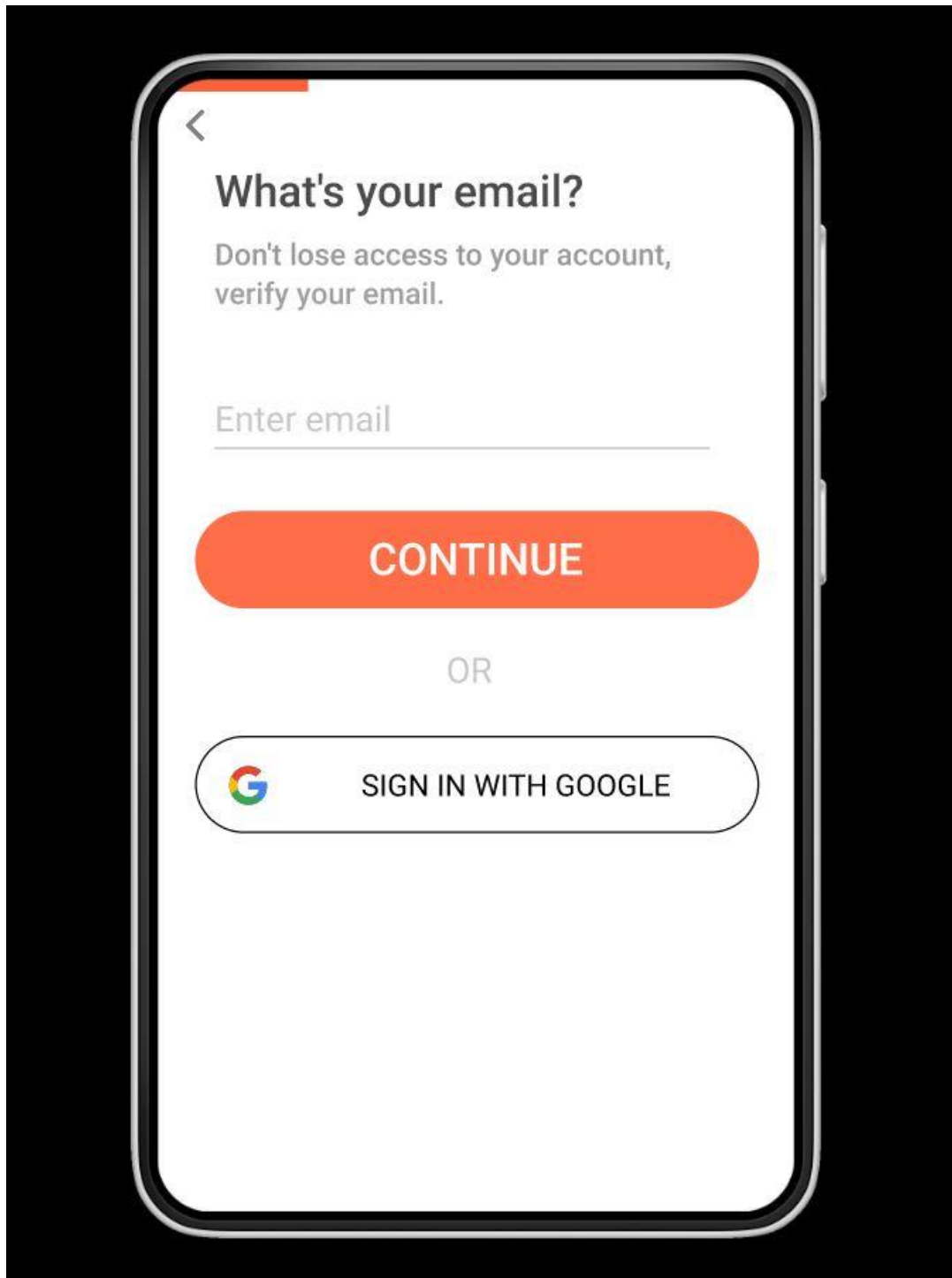


Figure 91.3: Unit Testing 1 Test Case

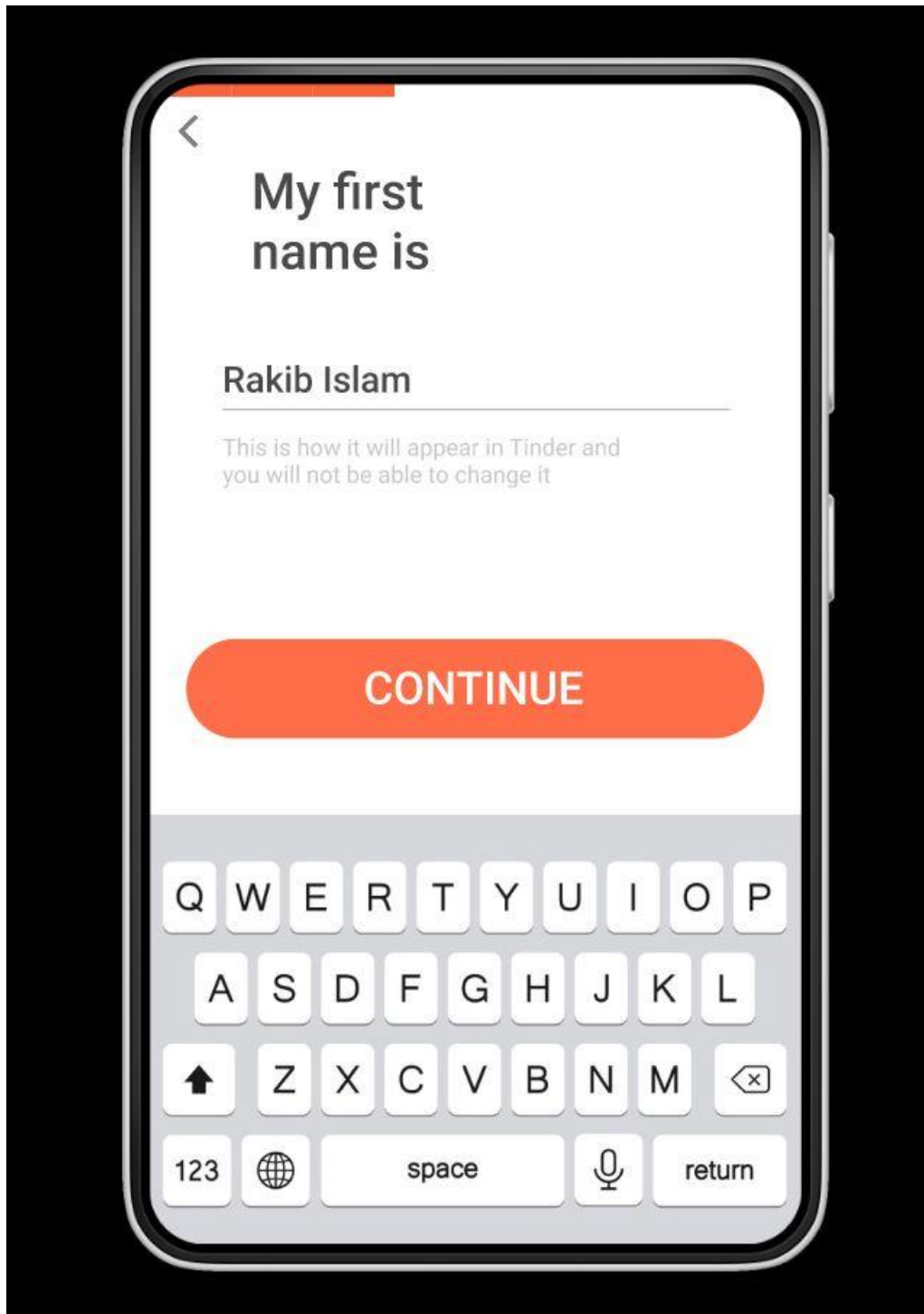


Figure 91.4: Unit Testing 1 Test Case

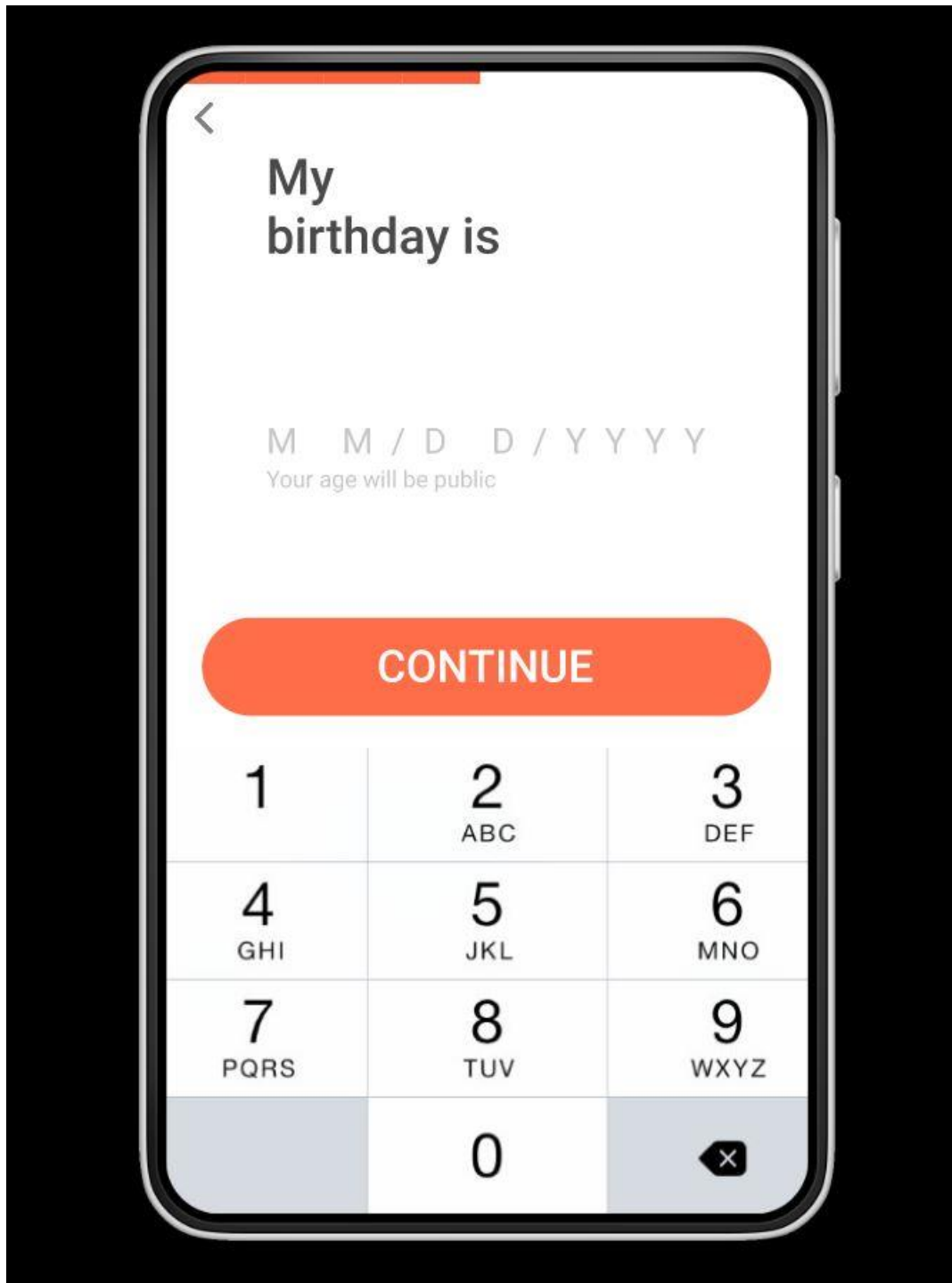


Figure 91.5: Unit Testing 1 Test Case

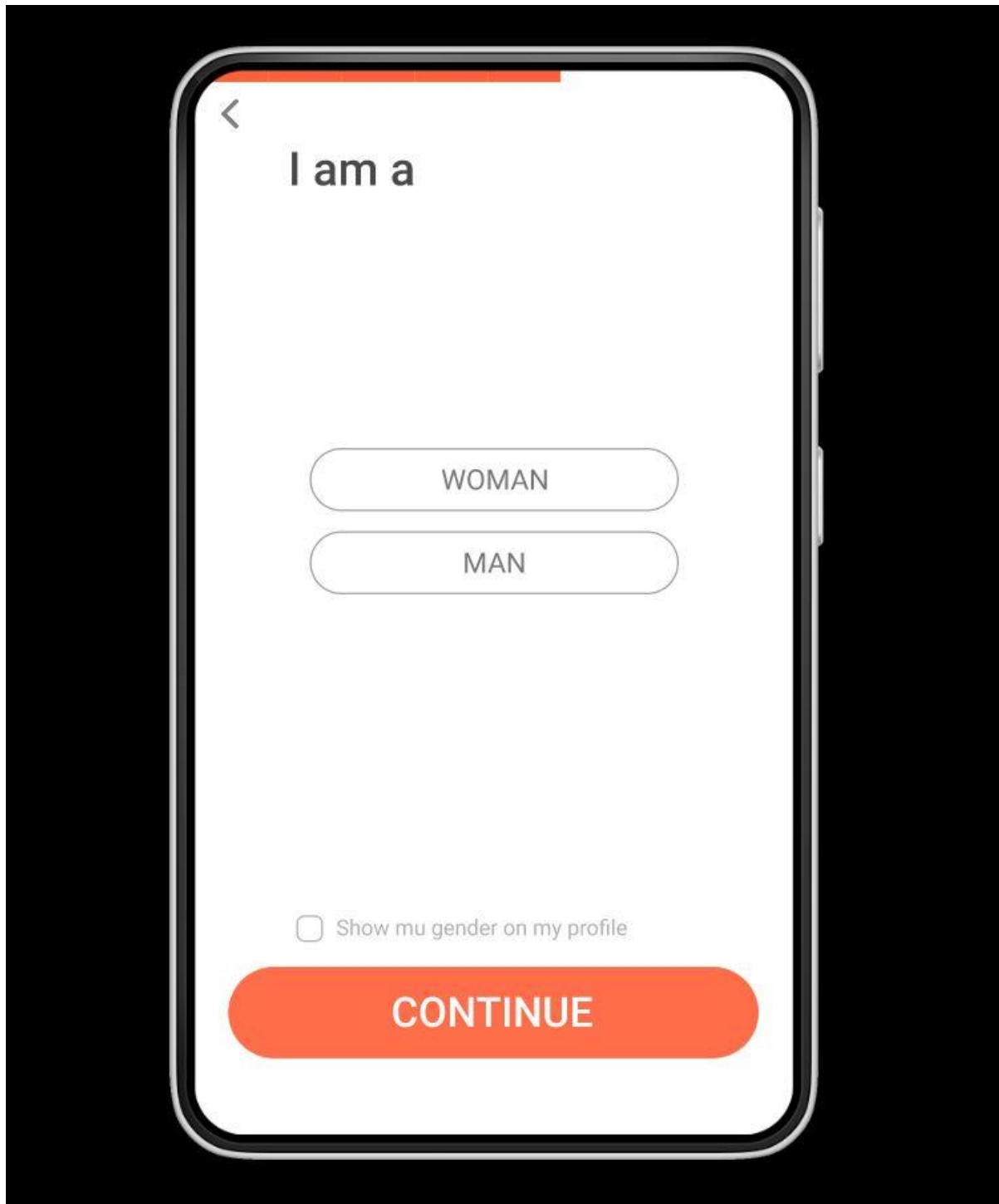


Figure 91.6: Unit Testing 1 Test Case

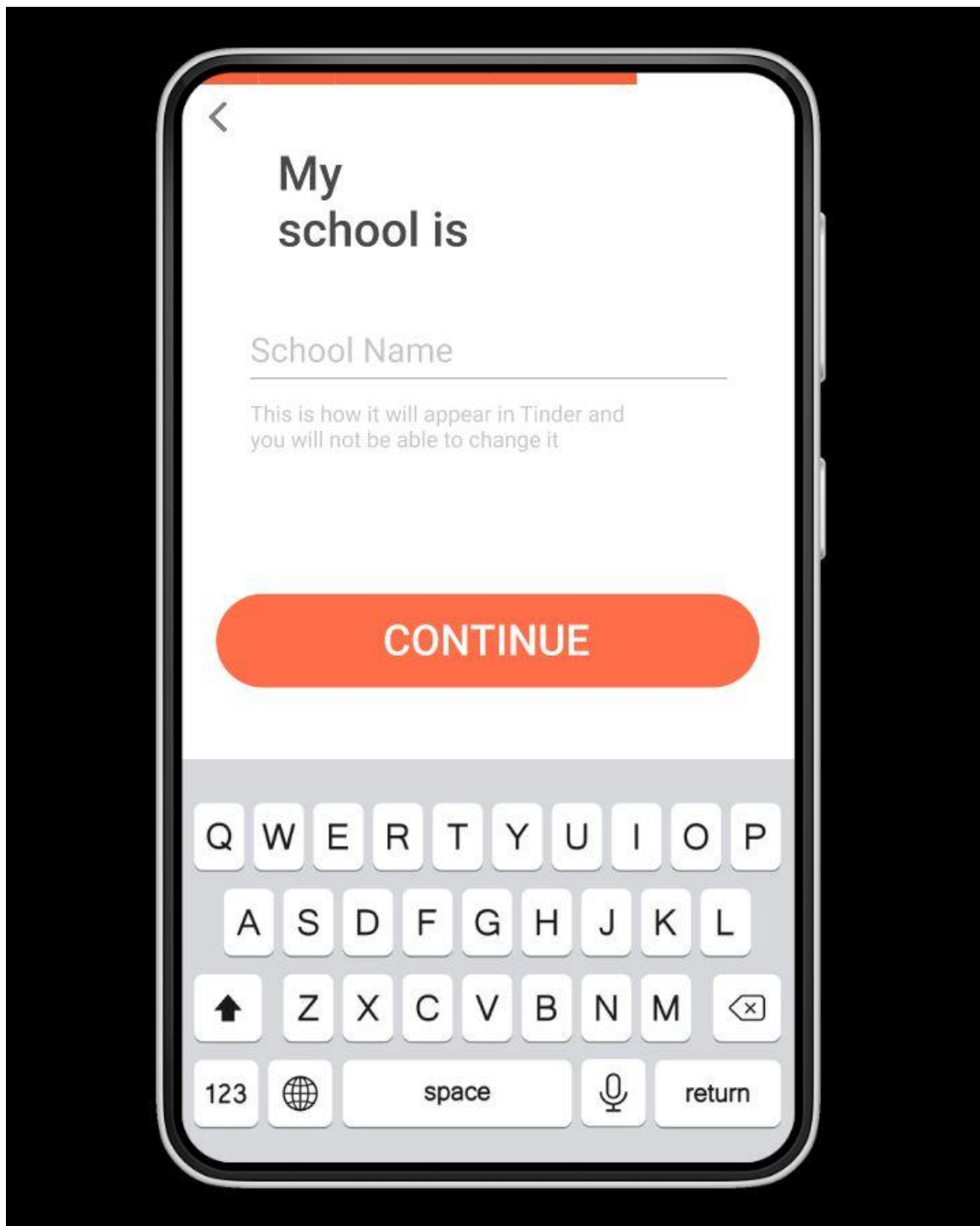


Figure 91.7: Unit Testing 1 Test Case

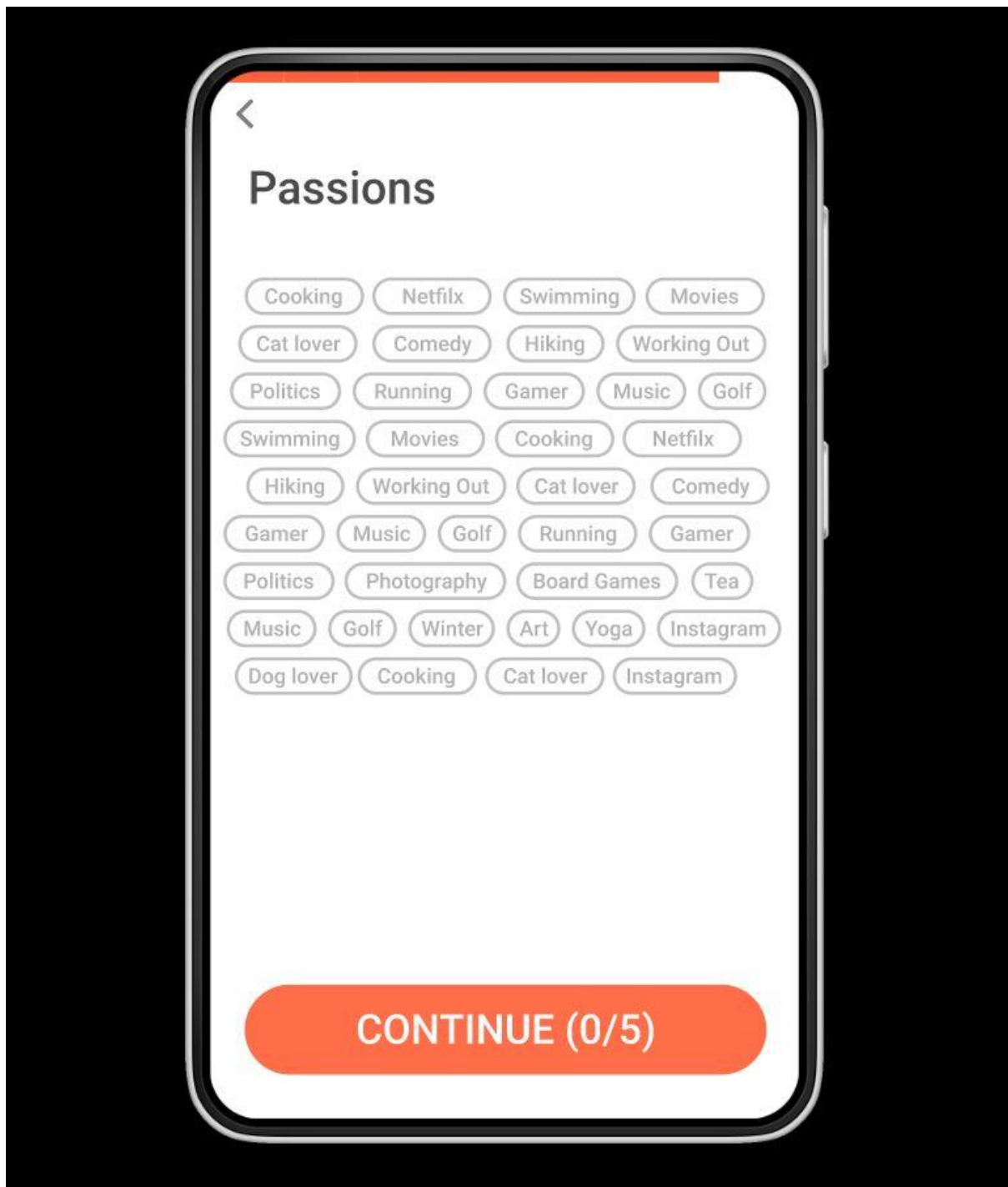


Figure 91.8: Unit Testing 1 Test Case

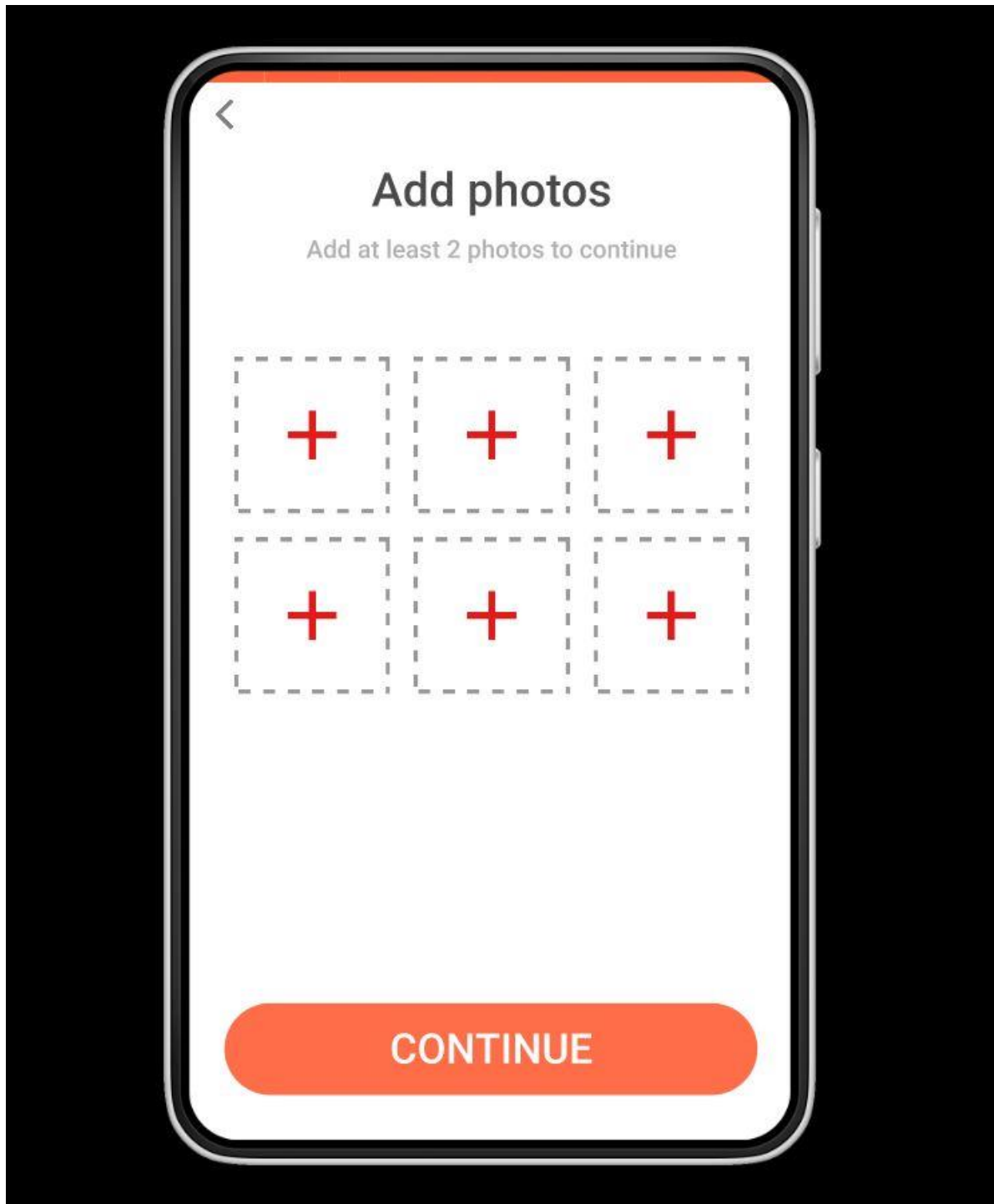


Figure 91.9: Unit Testing 1 Test Case

Unit Testing 2:

Test Case

Test Case Name	Unit Test		
Test Class	Customer Order Controller		
Test Description	Customer food order validation for customer order		
Data Source	Test Steps	Expected Result	Actual Result
	1. Select the order 2. select the payment system	An error message should return that customer order must be selected.	A message is showing that customer order is required for select payment method.

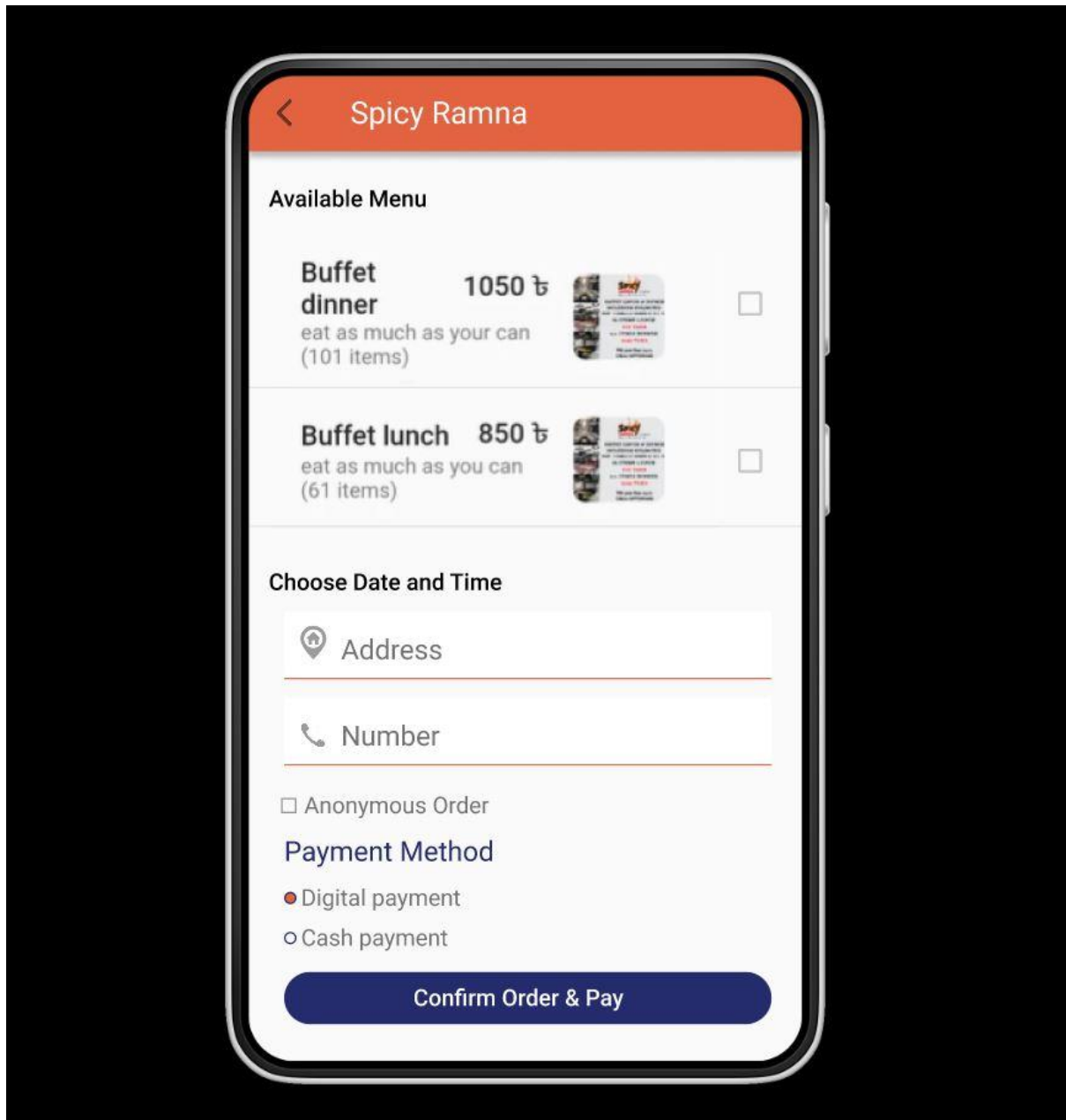


Figure 92: Unit Testing 2 Test Case

11.4 Module Testing

Module Test 1

Test Case

Test Case Name	Module Test		
Test Class	Customer\Restaurant's owner Controller		
Test Description	Restaurant's owner registration attempt without input.		
Data Source	Test Steps	Expected Result	Actual Result
	1. login as customer 2. click on add new restaurants	Error message should be displayed for all required fields.	The actual result displaying that the Customer name, email, and phone must not be empty.

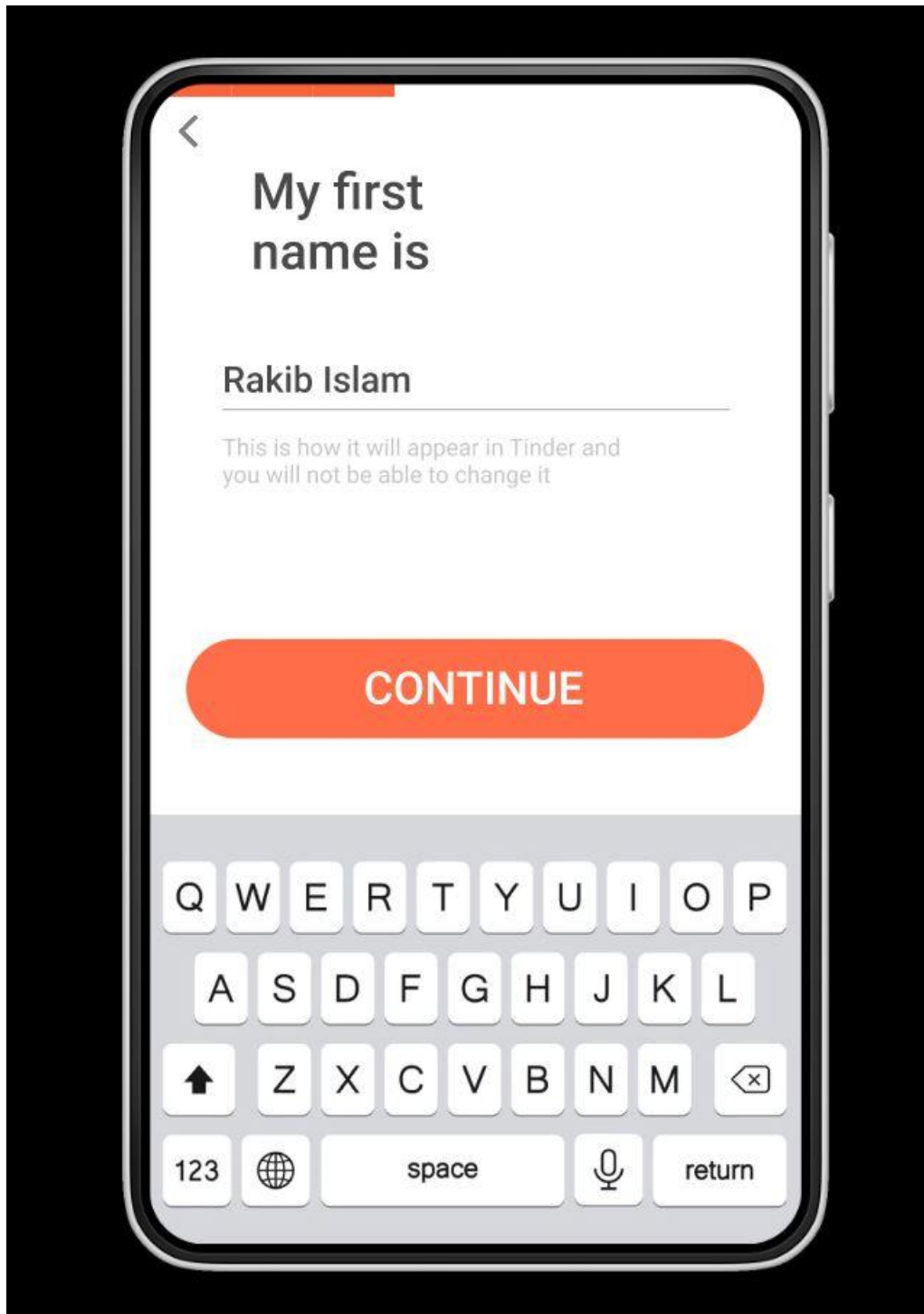


Figure 93: Module Testing 2 Test Case

11.5 Integration Testing

Integration Testing 1

Test Case

Test Case Name	Module Test		
Test Class	Delivery man Login Controller		
Test Description	Successful login attempt redirect.		
Data Source	Test Steps	Expected Result	Actual Result
	1.go to login page 2.provide valid credentials 3.press on login	The delivery person should be authorized and forwarded to the donor's residence after successful authentication.	It shows that email address is invalid

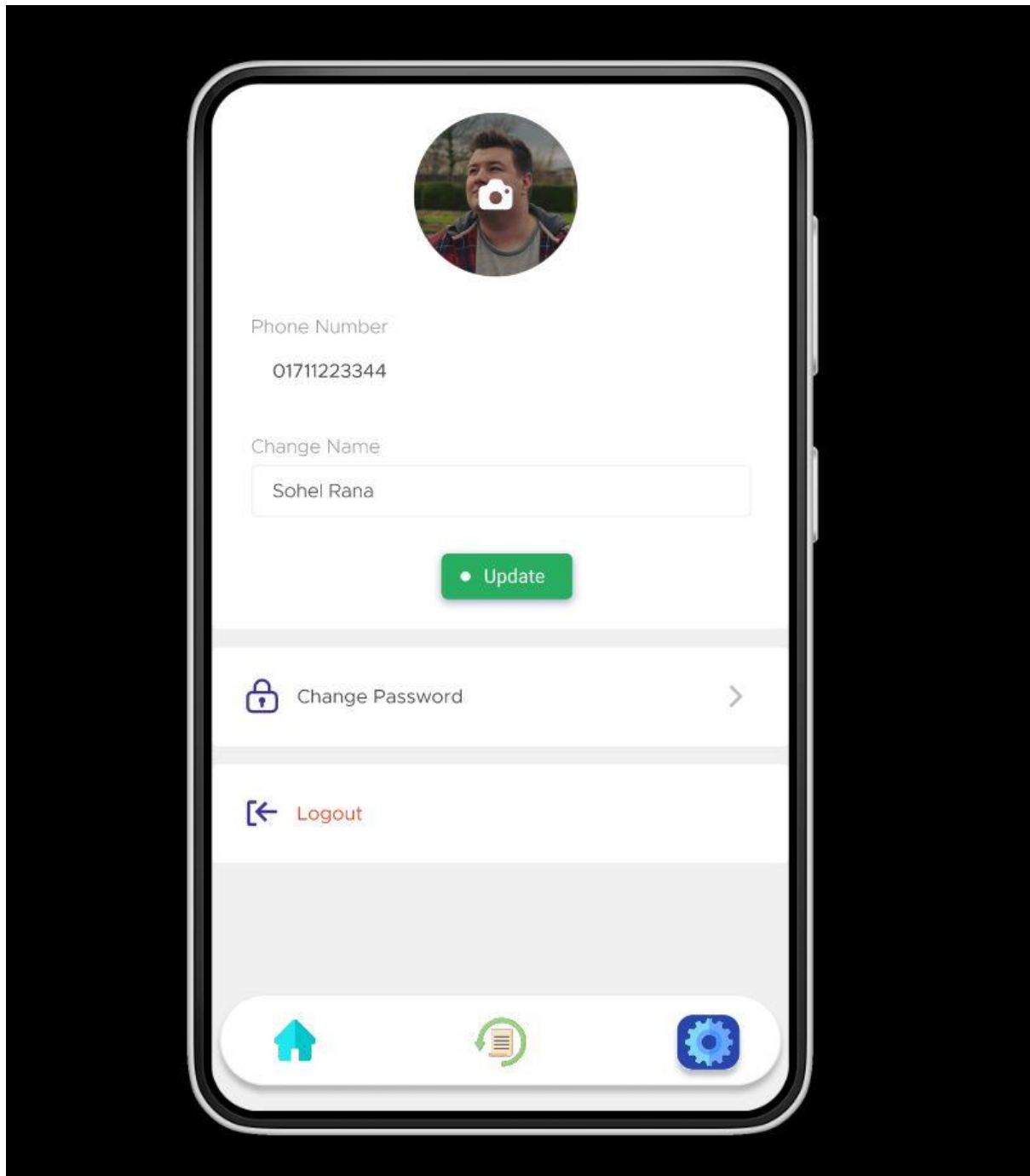


Figure 94: Integration Testing 2 Test Case

11.6 Security Testing

Security Testing 1

Test Case

Test Case Name	Module Test		
Test Class	Customer Login Controller		
Test Description	Invalid login attempt security testing.		
Data Source	Test Steps	Expected Result	Actual Result
	1.go to the login page 2.provide invalid credential and try to login.	If you are not logged in, a notice should appear.	The end outcome is exactly what you'd anticipate.

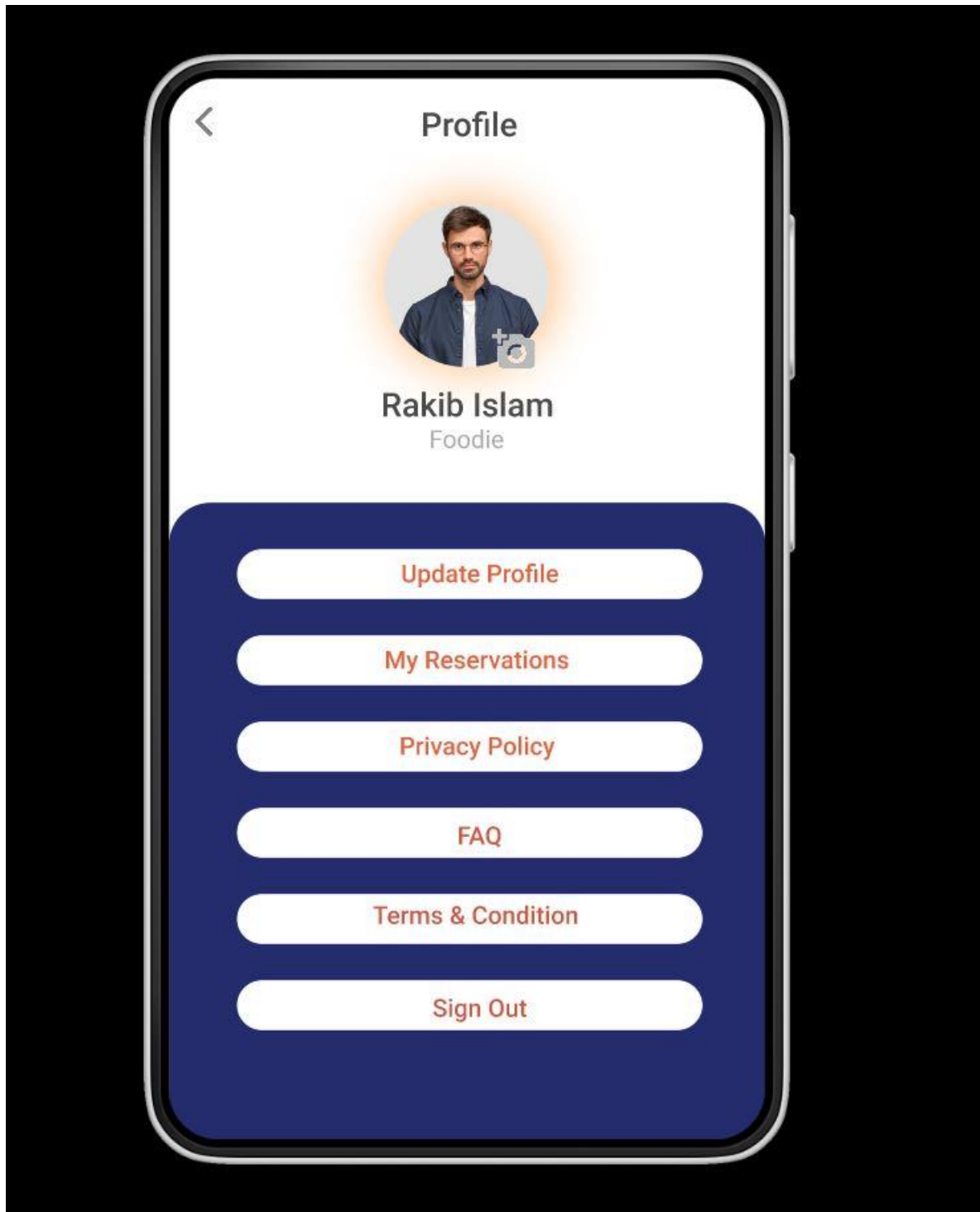


Figure 95: Security Testing 2 Test Case

Chapter 12 – Implementation

12.1 Training

Training is required to familiarize users with the newly designed system. The team must guarantee that the users or operators are capable of handling activities through the system once it is fully functional. Below is a diagram that depicts the training methods:

SL	User	Training Scope	Time Period	Comment
1	Restaurant's owner or Delivery man	Registering as a Restaurants owner, updating Order or delivery history, ensuring proper deliver food to the customer	2 Hours	The customer is grasping the procedures accordingly.
2	Admin	Add a database for the Customer need of food, acquiring restaurants owner for specific locations, confirming their actions properly.	1 Hour	Admin easily understands the operations.

Table 7: User training

12.2 Big Bang

Big bang works by quickly shutting off the old system and enabling the new one. This method is significantly faster than the others since it uses the new system right as the test is finished. The data transfer from the previous system may cause harm and instability, since data may be lost and the new system may crash. It is implemented on a single site using a single site's human resources.

12.3 Scaling (is there any plan?)

No, Because this project was granted to DCL, there are no plans to sell it.

12.4 Load Balancing

Load balancing refers to the optimization of a system against the effect of its users. The number of users struck indicates how many people are using the system at the same time and how long it has lasted. Load balance and load equalization are terms used to describe this. It distributes the demand over several servers, allowing the system to keep running smoothly. Because there are three different sorts of users and a large number of hits each day, a proper load balancing technique must be developed.

Chapter 13 – Critical Appraisal and Evaluation

13.1 Objective that could be met

The objectives which are initially declared that are listed below.

- Registration system for users
- Restaurant owner and Delivery man management
- Screening and transfusion record management
- Delivery man location tracking and eligibility checking.

Achievement rate and others

For all user kinds, the registration and login mechanism has been successfully deployed. Customers, restaurant owners, and delivery drivers may complete the registration procedure and then log in to the system. For authenticating and authorizing users to the system, I utilized the standard Laravel authentication mechanism. The rate of success is 100 percent.

13.2 Objective that totally don't meet of touched

For the actions, practically the full article is consulted. However, there is one target that has not connected the mobile application for tracking the whereabouts.

The reasons why it could not be touch

The main reason for not being contacted about the aim is a lack of time to complete the tasks. It would be quite difficult due of the short time frame, and another issue is the requirement for a continual facilitating server, which is also highly difficult for the work. Because the program cannot function without a live server, it cannot evolve.

Chapter 14 – Lessons Learned

14.1 Pre project – review – closing

To build up the Foodpatrons System mainly a mobile application with android location tracking, from the start I need to follow a predefined structure like project proposal submitting, project title defense, development and documentation. The system mainly concerned about the food and dating opportunity.

14.2 Critical appraisal

In Bangladesh, there is no app that allows users to locate a random companion to hang out with in a restaurant, as well as book reservations and place food orders. Food delivery services such as Foodpanda, Sohoz Food, HungryNaki, Pathao Food, and others have played a vital role in this market during the previous five years. Social networking, on the other hand, is a great way to meet new people and make friends. On the other hand, several blind dating apps, such as Tinder, are playing an increasingly important role in blind dating. People are adapting numerous cultures in this era of globalization. Younger generations, in particular, are seizing platform-based entertainment alternatives. As a result, there is a growing need for this sort of platform.

14.3 What have I learned

During this report I have learned so many things especially how to make the project report, what are the major parts of a report etc. Most importantly working on designing a project UI from sketch to final prototype was very knowledgeable. I have learned some UI/UX designing tool and Microsoft word and PowerPoint. Making this report also help me to manage some interpersonal qualities like managing stress, improving patience etc. This report helps me to learn so many things that cannot be described in words.

14.4 What problem I have faced

During making this project, the major problem I faced is time limitation. Completing the project report within very short time was very challenging. Another problem was lack of knowledge. Without sound knowledge of making such big report, it is difficult to make the whole report. Individually making the full report was really challenging.

14.5 What solutions occurred

This report was made by following the instructions of my supervisor that was really a great support. To overcome the time limitation, I have prepared this report in very brief and to the point. I've also followed a report making template provided by my supervisor that was very helpful to complete this report.

Chapter 15 – Conclusion

15.1 Summary of the project

Foodpatrons is a platform-based food sharing and dating platform where people can invite random people to have food together, book restaurants, and also get food delivery. In today's world digital platform is very popular among all age group. People

are very busy now a days. They hardly get time to find their enjoyment. Moreover, people are very busy this platform will help to find partners, book restaurants etc.

15.2 Goal of the project

The main goal of this project is connecting people. To provide entertainment facilities within shortest possible time.

15.3 Success of the project

The success of the project will be achieved if the users get the benefit from this platform. If the security is properly ensured and all enlisted partners/restaurants of this platform can server properly.

15.4 What I have done in the documentation (stages / activities / plans)

I have learned so many things in documentation part especially how to make the project report, what are the major parts of a report etc. Most importantly working on designing a project UI from sketch to final prototype was very knowledgeable.

15.5 Value of the project

This project will be highly valuable in terms of connectivity and engagement. The commercial aspects of this project is also high. Specially the market size is very potential and attractive. The restaurant and food delivery business is very booming in today's world. Moreover, social media and connectivity platform is also demanding. Dating platforms are also very trendy in all countries around the world. Foodpatrons will be the single platform for all facilities together.

15.6 My experience

Working on such a big project is very exciting and challenging. I have learned many theoretical and practical knowledge especially the UI and UX design part, I have learned design principal, designing tools etc. In the documentation part I have learned formal report writing tactics, Use case diagram development skill comprehensive writing skills etc. Overall, this report helps me to learn very in-depth of a software development project.

Appendix:

Plagiarism Report

Turnitin Originality Report

Processed on: 21-Feb-2022 14:06 +06
 ID: 1767437634
 Word Count: 9367
 Submitted: 1

181-16-231 -- Fattahul Islam.pdf By Fattahul Islam

Similarity Index	Similarity by Source	
27%	Internet Sources:	14%
	Publications:	0%
	Student Papers:	25%

10% match (student papers from 29-Dec-2019)

Class: Final_project_Fall2019

Assignment: Project_Fall_2019

Paper ID: [1238624034](#)

6% match (student papers from 18-Dec-2015)

Submitted to Mountbatten Internship Program on 2015-12-18

4% match (student papers from 03-Apr-2018)

Submitted to Daffodil International University on 2018-04-03

2% match (student papers from 18-Dec-2017)

Submitted to University of Greenwich on 2017-12-18

1% match (student papers from 04-Jan-2016)

Submitted to University of Greenwich on 2016-01-04

1% match (student papers from 30-May-2017)

Submitted to University of Greenwich on 2017-05-30

1% match (Internet from 15-Jan-2022)

<https://www.coursehero.com/file/115122459/A-case-study-on-krishi-bankdocx/>

1% match (student papers from 19-Aug-2020)

Submitted to NCC Education on 2020-08-19

< 1% match (student papers from 03-Apr-2018)

Submitted to Daffodil International University on 2018-04-03

< 1% match (student papers from 28-Apr-2016)

Submitted to University of Greenwich on 2016-04-28