Deliverable 4 Final Report

Course code: SEG 2105
Professor: Wassim El Ahmar
Group #36

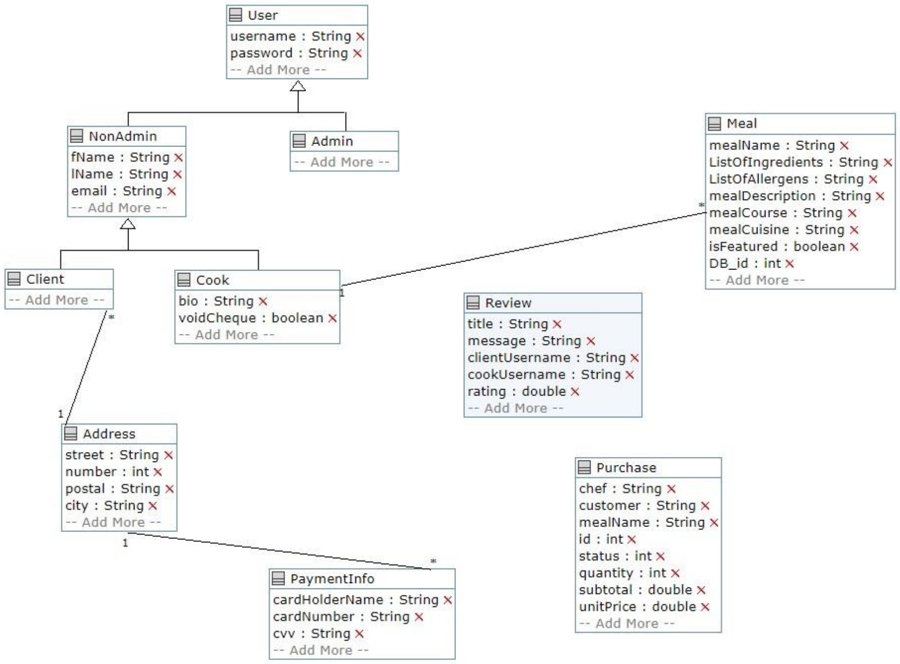
Georgin Binoy: 300233721
Fuad Thabet: 300255031
Steven Mackenzie Sajder: 8216307

Due date: Dec/7/21

Introduction:

Our objective was to develop a comprehensive and practical Ottawa-based meal sharing mobile application. This application supports three types of users, Cooks, Clients and the Administrator. The cooks are allowed to create meals to be featured on the main menu that is accessible for clients to purchase. The clients then choose between a variety of options from all cultures of cuisines and are encouraged to leave reviews. The administrators main priority is to take the negative complaints that customers leave and decide whether the cooks responsible should be suspended, banned, or given another chance. We were tasked to create features that would allow all three users to seamlessly operate the application from scratch. Our main inspirations throughout the project were UberEats, DoorDash, HelloFresh and SkipTheDishes. We utilized core GUI techniques from these apps and refactored their design principles for an original and simplistic interface. Over the last couple months our vision and goals were set to create the best application yet and we settled for nothing less than the top. We hope you enjoy our application as much as we did building it.

UML Diagram



Contributions

Deliverable 1

Features	
The team created in GitHub contains all members	Georgin, Fuad, Mac
Each member of the group has made at least one commit to the repository.	Georgin, Fuad, Mac
UML Class diagram of your domain model	Fuad
A user can create Client or Cook account	Georgin
The administrator, cook, client user can see the welcome screen after successful authentication	Faud, Mac
The user can log off	Fuad, Mac
All fields are validated	Fuad

Deliverable 2

Features	
Updated Uml Class diagram of your domain model	Fuad
4 unit test cases	Georgin
When a user registers, their account is stored in the DB	Fuad

The admin can view the list of complaints	Fuad
The Administrator Can action The complaint (dismiss complaint Or suspend cook) For a temporary suspension, the Administrator the date when the suspension can be lifted The Complaint disappears from the Administrator’slist once it is actioned The complaints list is stored in the DB	Faud, Mac
The Cook can see a message informing them that they have been suspended For temporary suspension,the Message informs them when The suspension will be lifted For permanent suspension, the message informs them that they can No longer use the application	Fuad

Deliverable 3

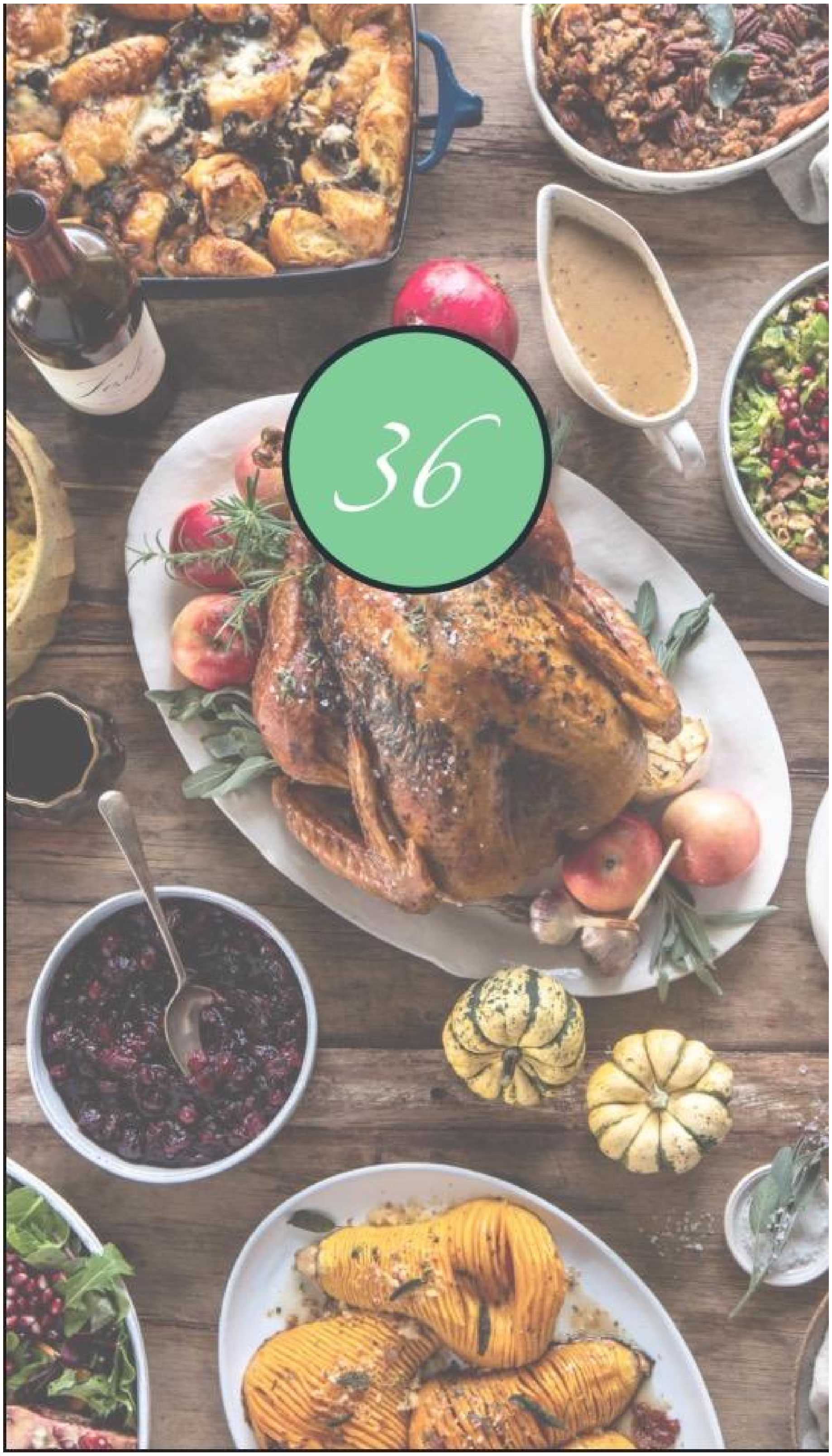
Features	
UML Diagram	**Fuad**
4 unit tests cases	**Mac**
Cook can add a meal to menu	**Mac**
Cook can add a meal to the offered meals list	**Fuad**
Cook can delete from the menu	**Mac**
Cook can't delete from menu if it is a featured meal in the menu	**Fuad**
The cook can remove a meal	**Georgin**
When a suspended cook logs on only see the suspension	**Fuad**
All fields are validated and appropriate	**Fuad**

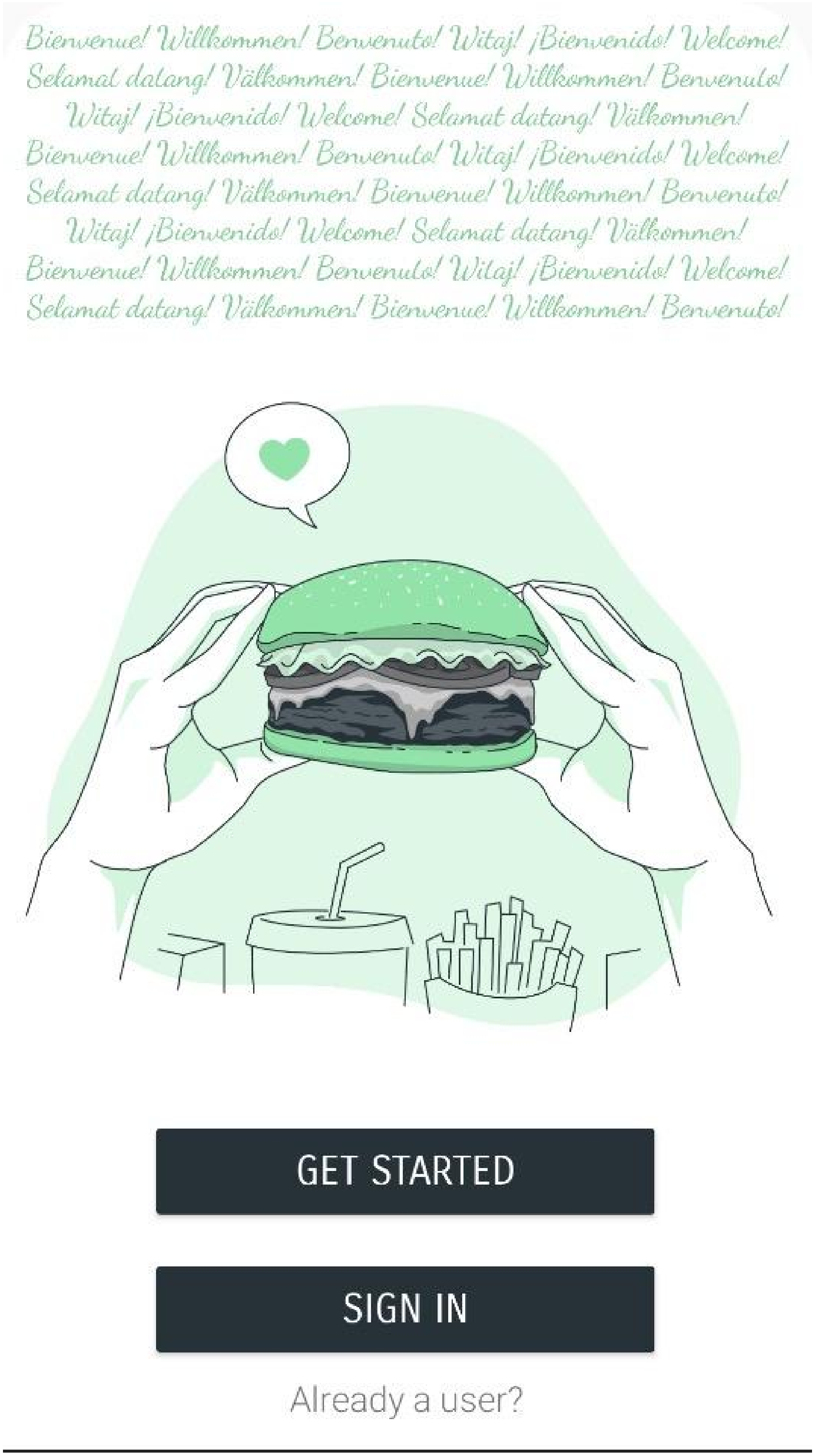
messages show up	

Deliverable 4

Features	
UML Diagram	**Fuad**
4 unit tests	**Georgin**
Final report	**Georgin**
Client can search for a meal and the client can see search results for meals offered by non-suspended cooks	**Mac, Fuad**
The client can view the cooks info and rating for every meal in the search result. Also the client can view the meal’s information for each meal	**Fuad**
The client can submit a purchase request and the cook can receive a purchase request submitted by the client	**Fuad**
The client can view the status of their purchase (pending, approved, rejected)	**Mac**
The client can rate the cook from which they can have purchased a meal	**Georgin**
The client can submit a complaint about the cook to the admin	**Mac**
The cook can view and approve/reject purchase requests received by clients.	**Faud**
The cook can view their profile rating	**Fuad**
All fields should be validated	**Fuad**

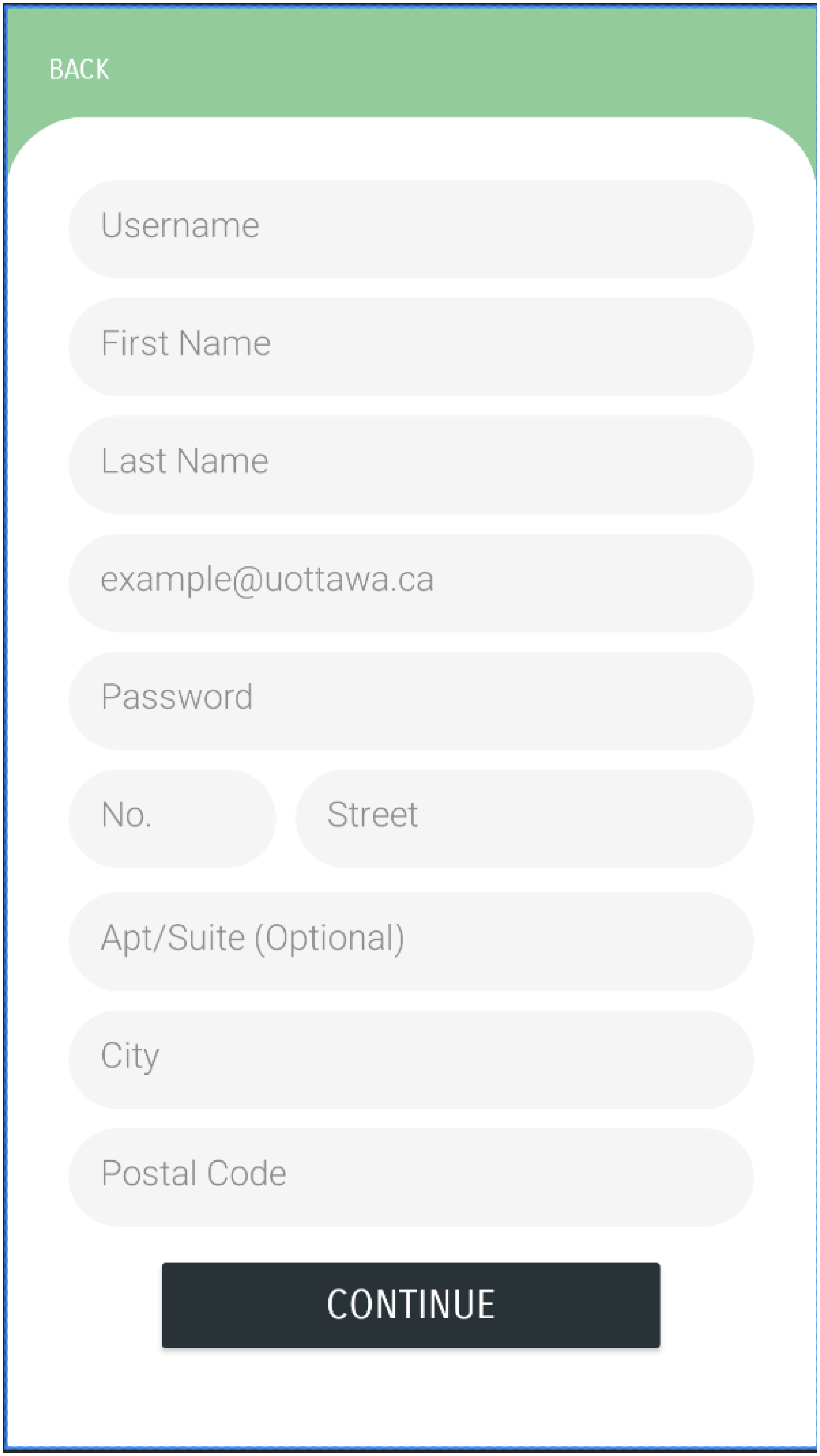
Screenshots

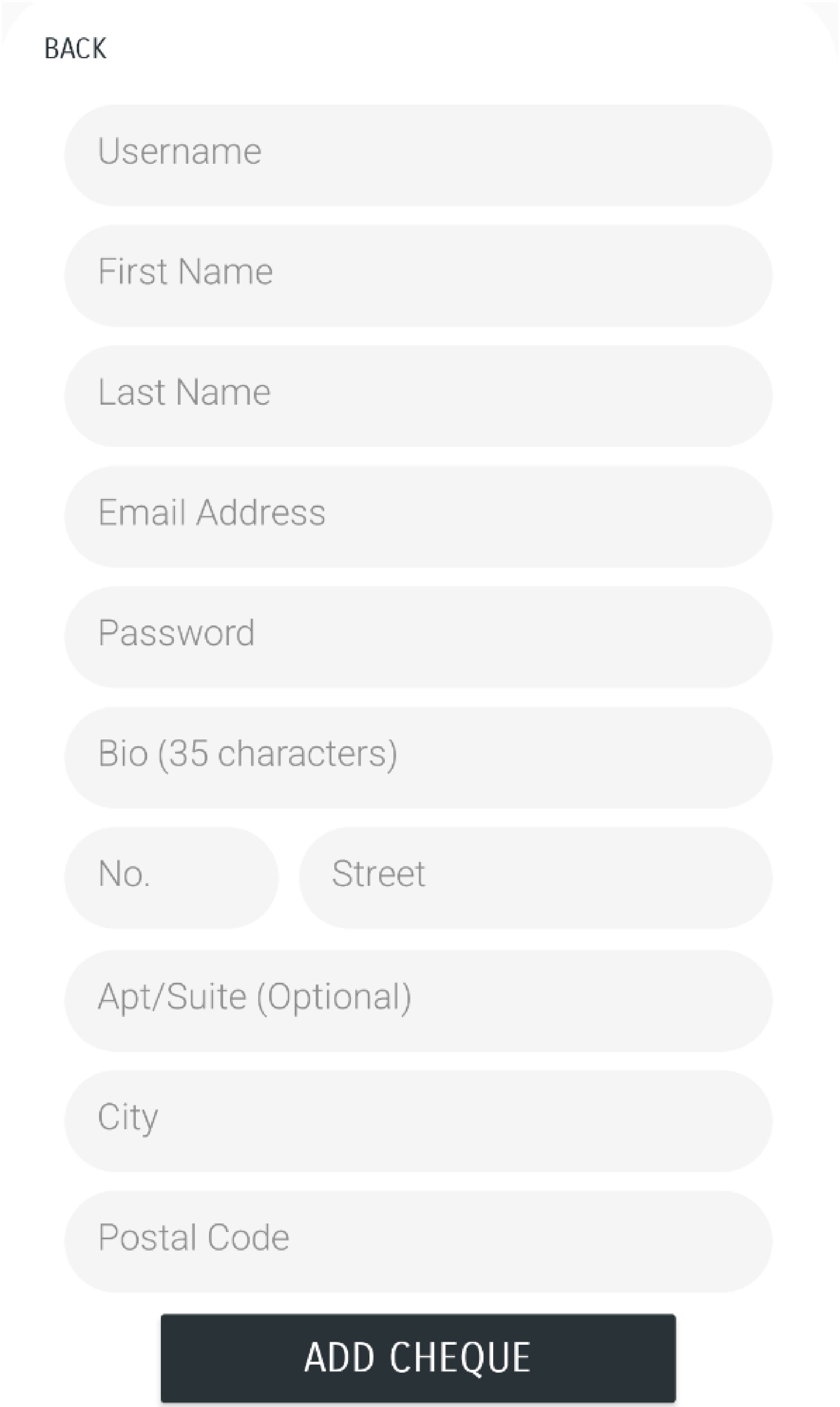



Bienvenue! Willkommen! Benvenuto! Witaj! ¡Bienvenido! Welcome!
Selamat datang! Välkommen! Bienvenue! Willkommen! Benvenuto!
Witaj! ¡Bienvenido! Welcome! Selamat datang! Välkommen!
Bienvenue! Willkommen! Benvenuto! Witaj! ¡Bienvenido! Welcome!
Selamat datang! Välkommen! Bienvenue! Willkommen! Benvenuto!
Witaj! ¡Bienvenido! Welcome! Selamat datang! Välkommen!
Bienvenue! Willkommen! Benvenuto! Witaj! ¡Bienvenido! Welcome!
Selamat datang! Välkommen! Bienvenue! Willkommen! Benvenuto!
GET STARTED
SIGN IN
Already a user?

BACK

Cook or Eat?


BACK
Username
First Name
Last Name
example@uottawa.ca
Password
No.
Street
Apt/Suite (Optional)
City
Postal Code
CONTINUE



Username

First Name

Last Name

Email Address

Password

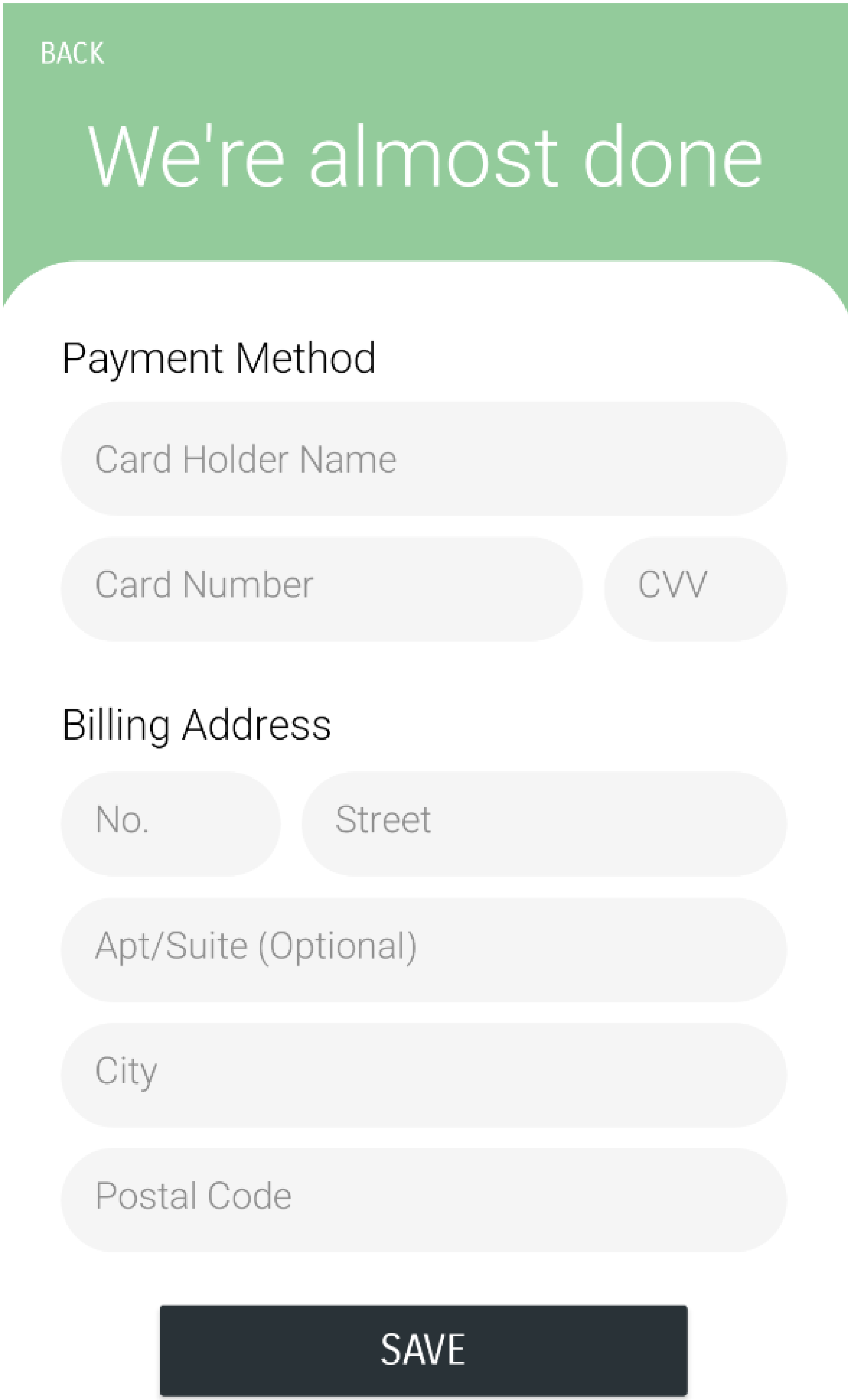
Bio (35 characters)

No. Street

Apt/Suite (Optional)

City

Postal Code



BACK

We're almost done

Payment Method

Card Holder Name

Card Number

CVV

Billing Address

No.

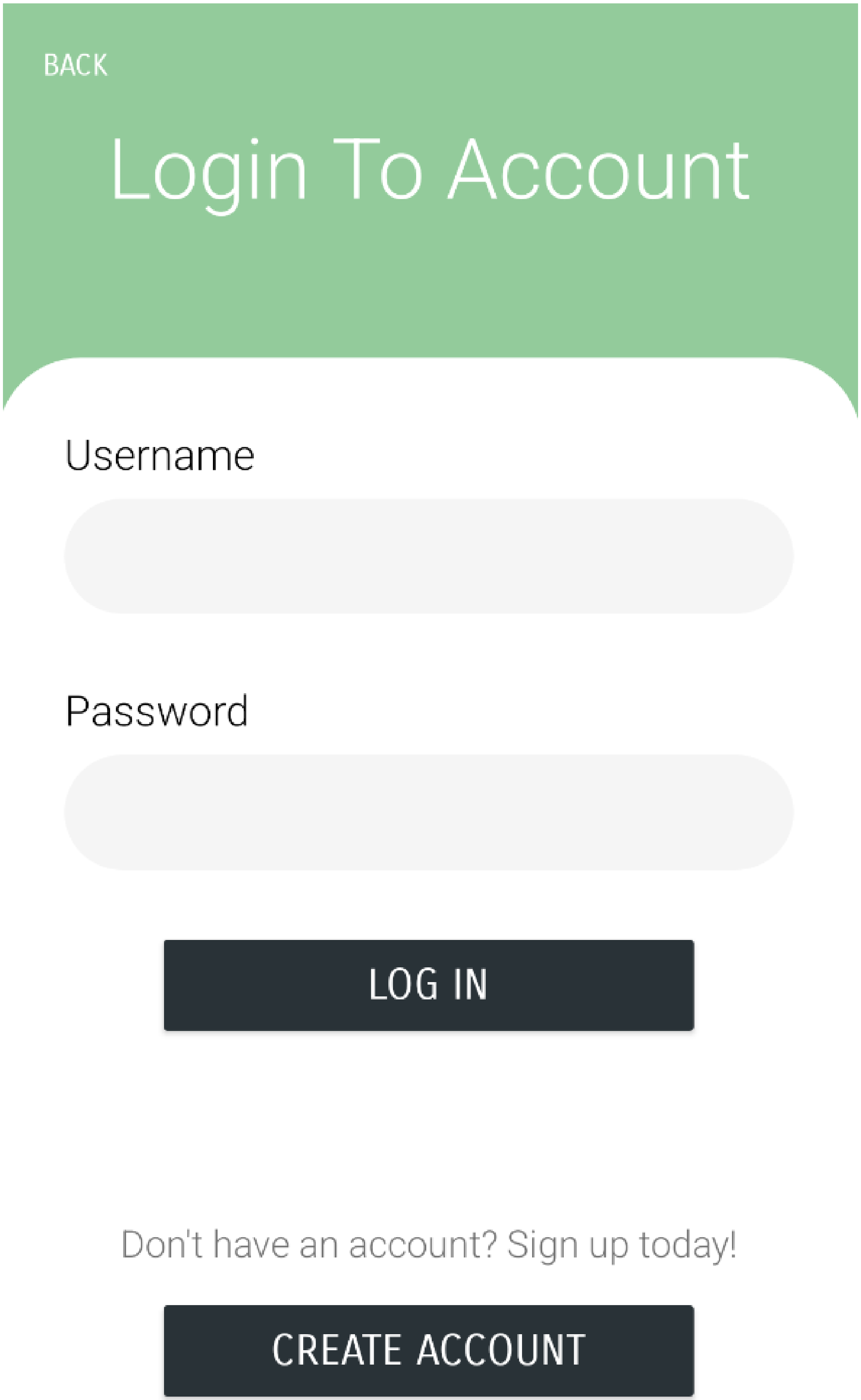
Street

Apt/Suite (Optional)

City

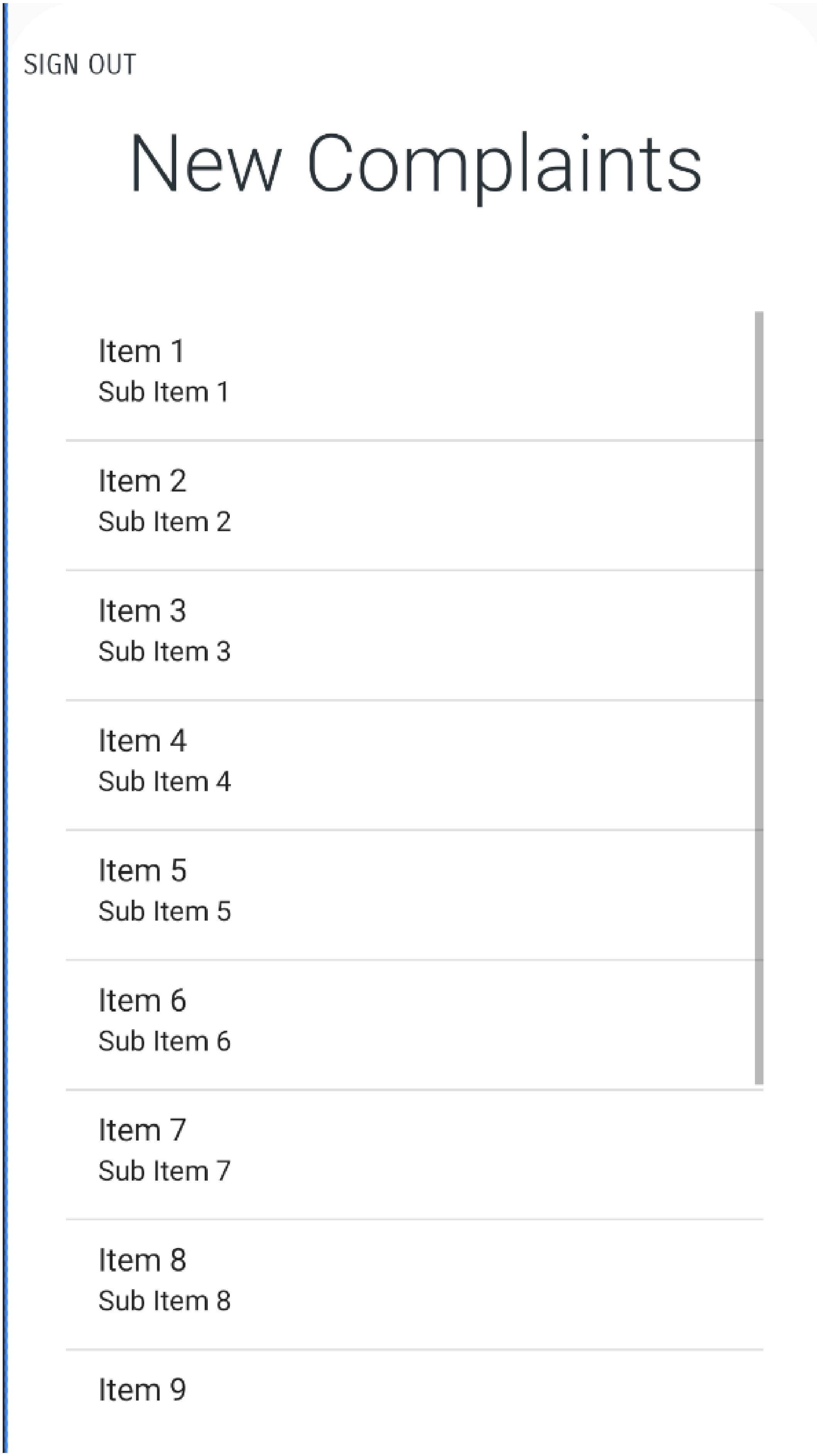
Postal Code

SAVE



Login To Account

Don't have an account? Sign up today!



New Complaints

Item 1
Sub Item 1

Item 2
Sub Item 2

Item 3
Sub Item 3

Item 4
Sub Item 4

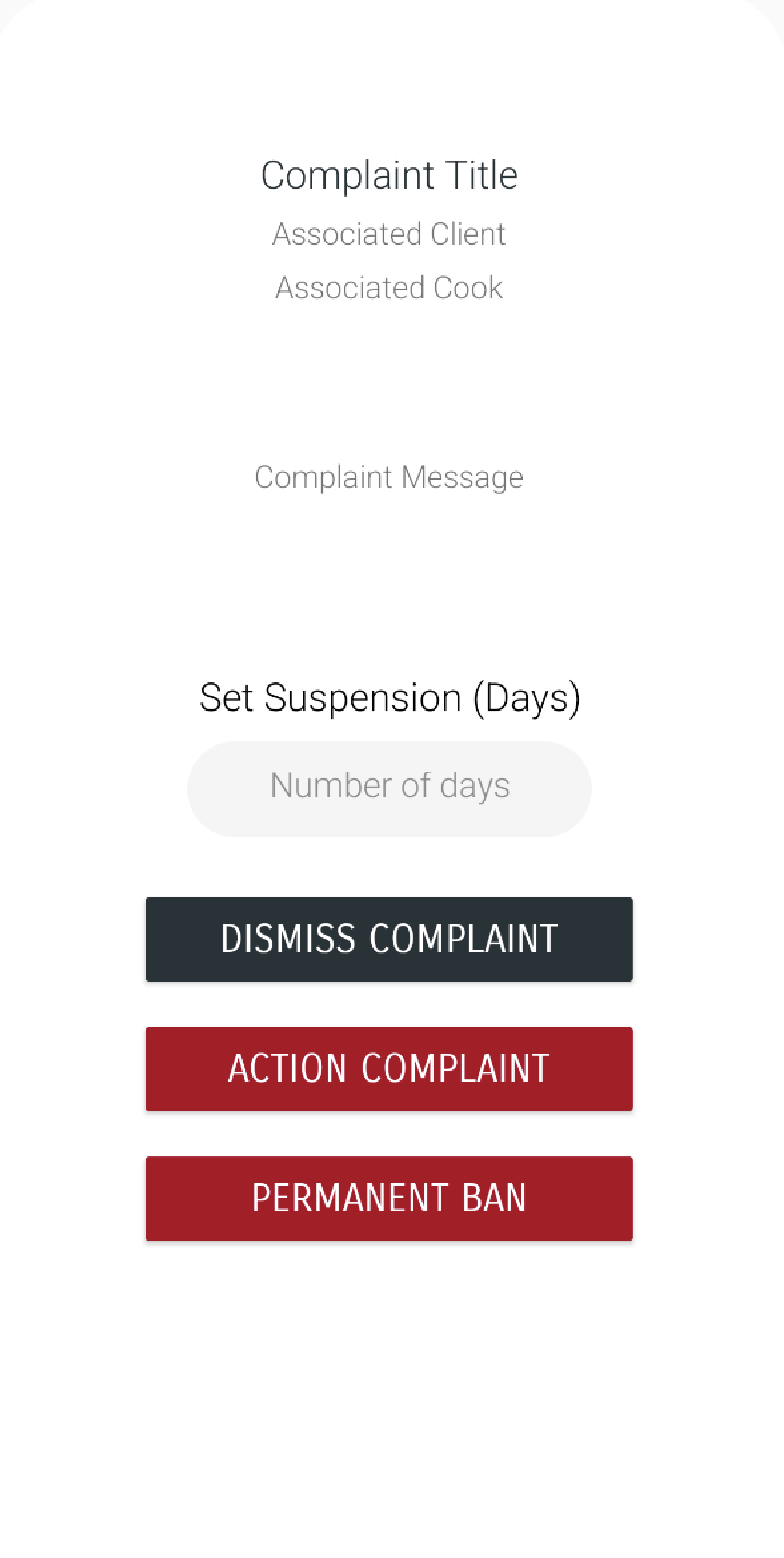
Item 5
Sub Item 5

Item 6
Sub Item 6

Item 7
Sub Item 7

Item 8
Sub Item 8

Item 9



Complaint Title

Associated Client

Associated Cook

Complaint Message

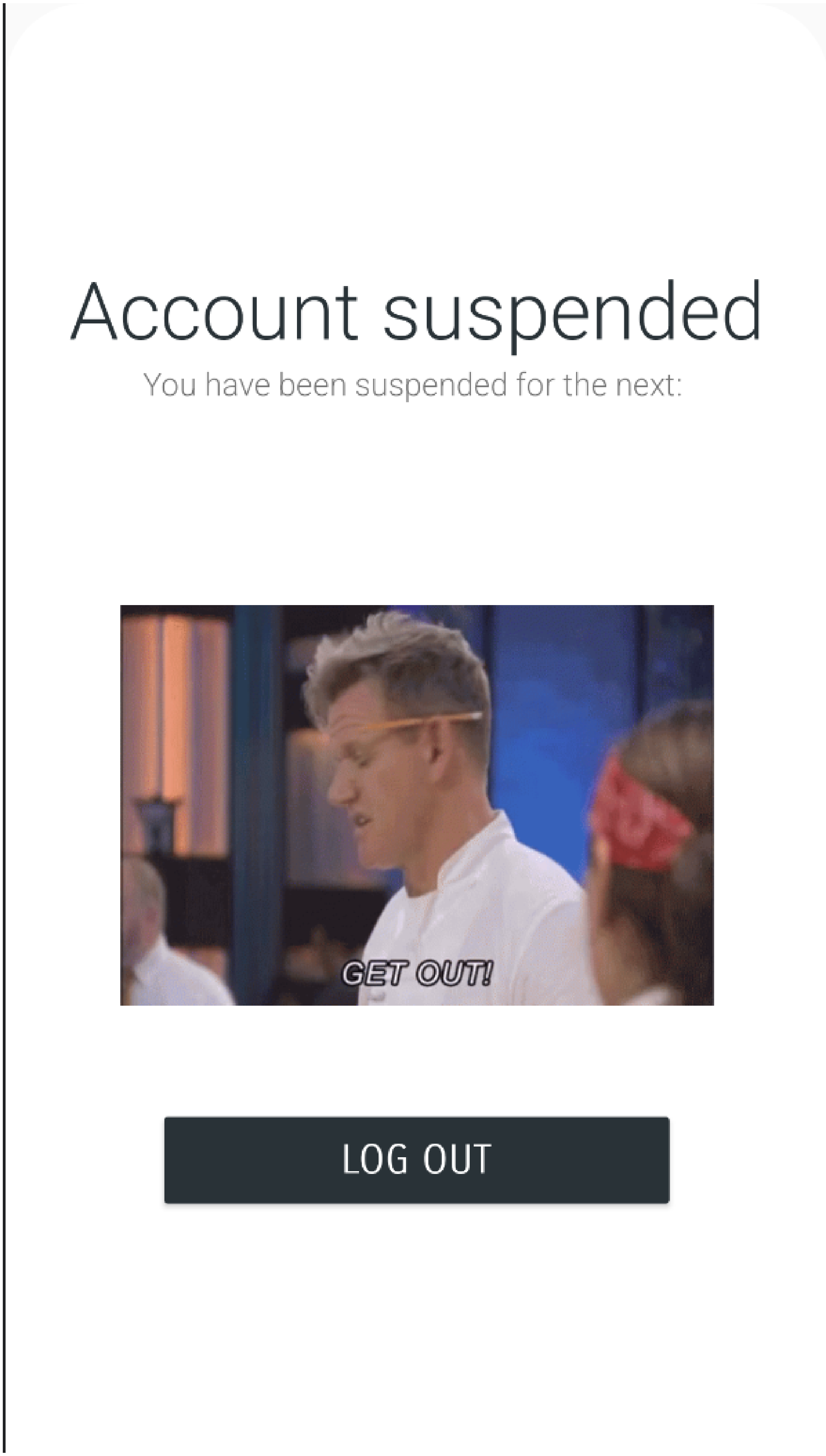
Set Suspension (Days)

Number of days

DISMISS COMPLAINT

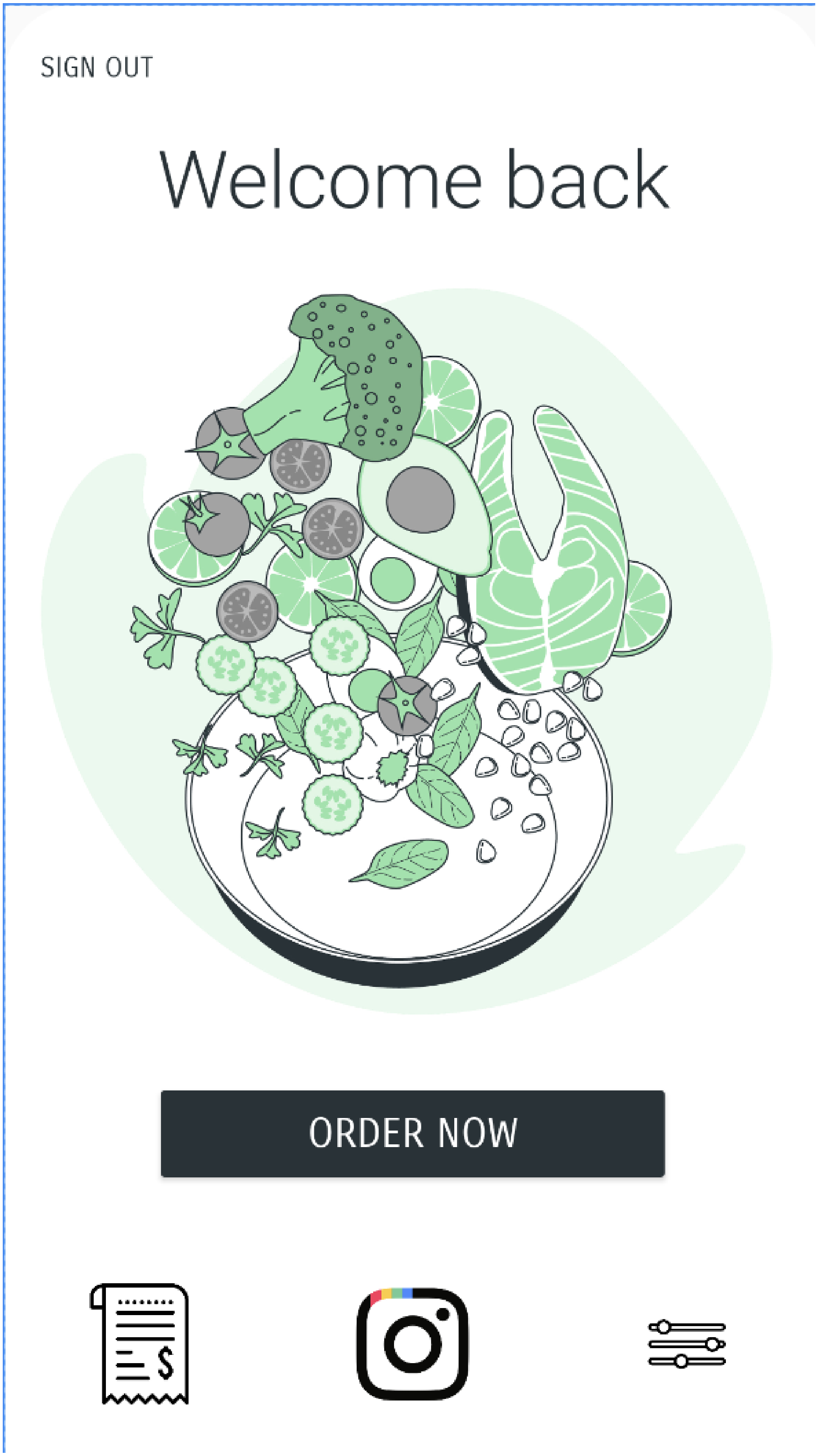
ACTION COMPLAINT

PERMANENT BAN

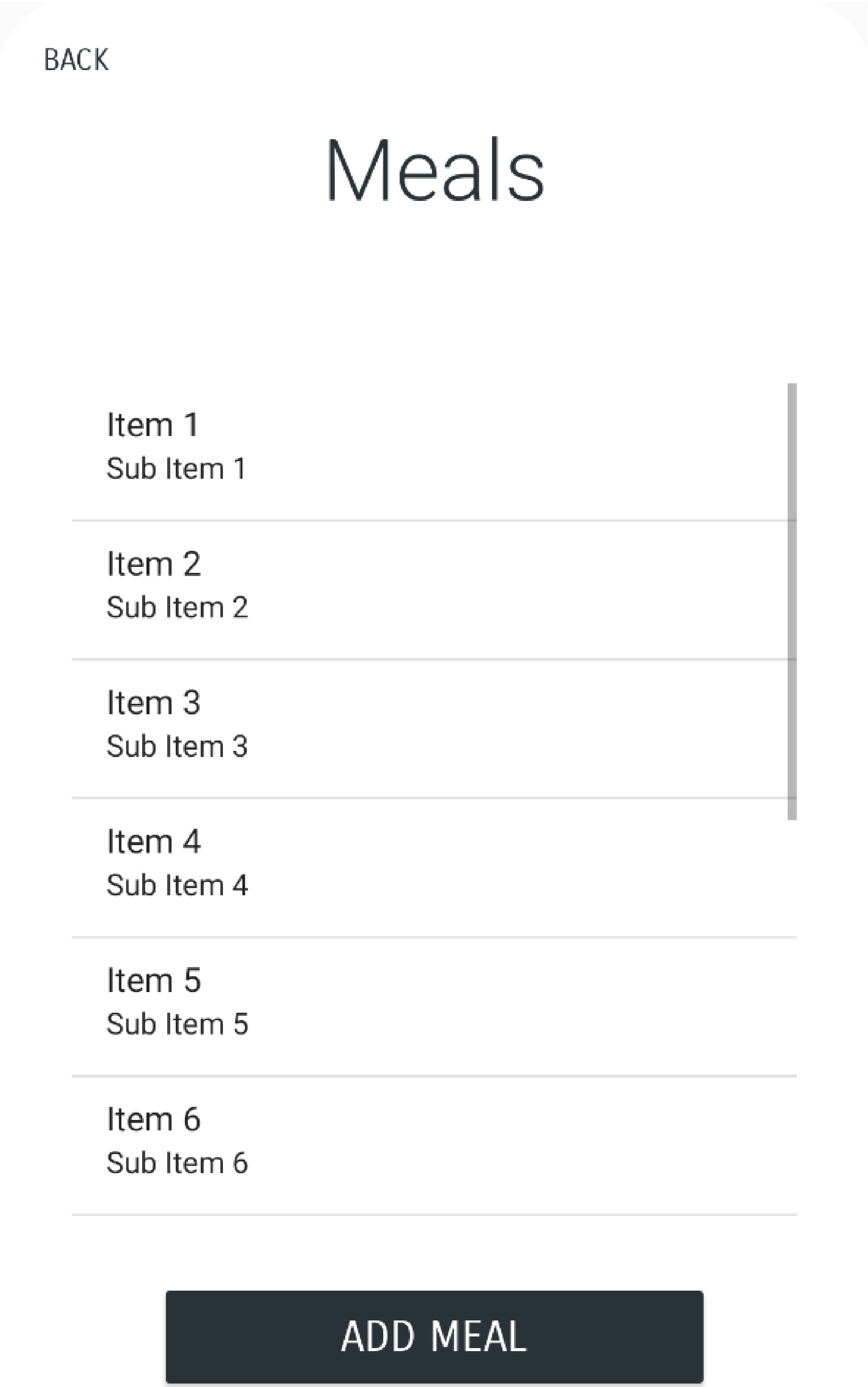


Account suspended

You have been suspended for the next:



Welcome back



Meals

Item 1
Sub Item 1

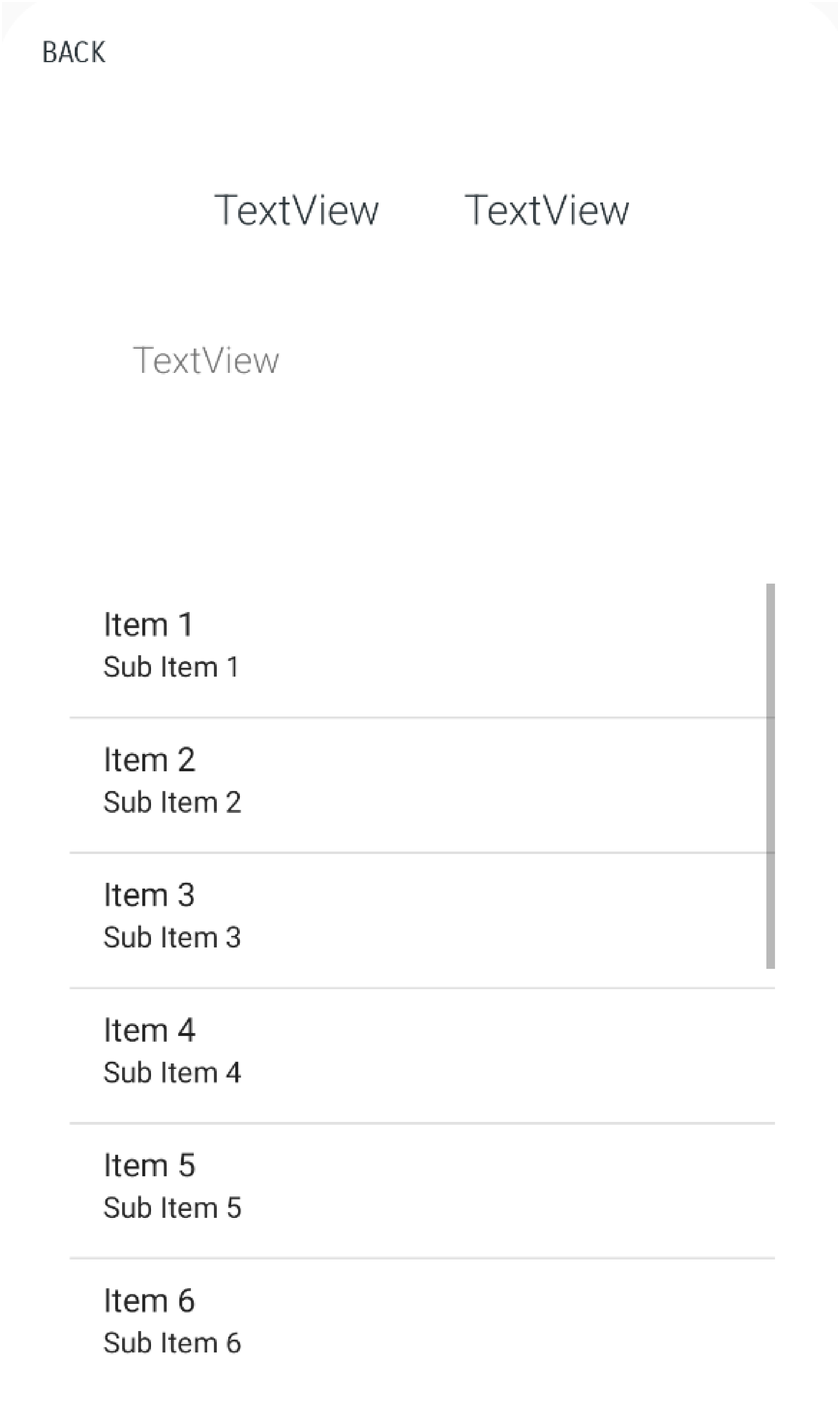
Item 2
Sub Item 2

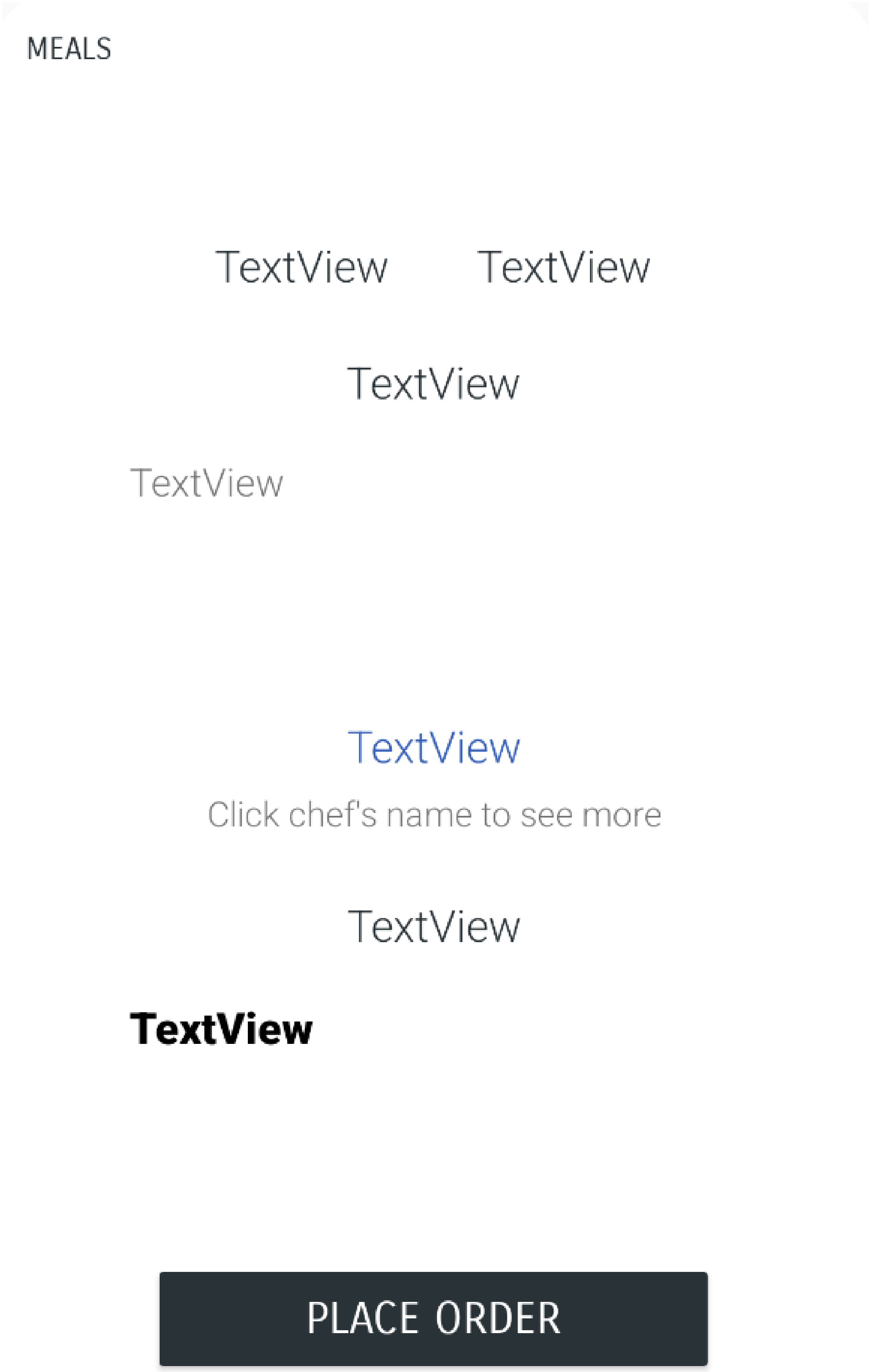
Item 3
Sub Item 3

Item 4
Sub Item 4

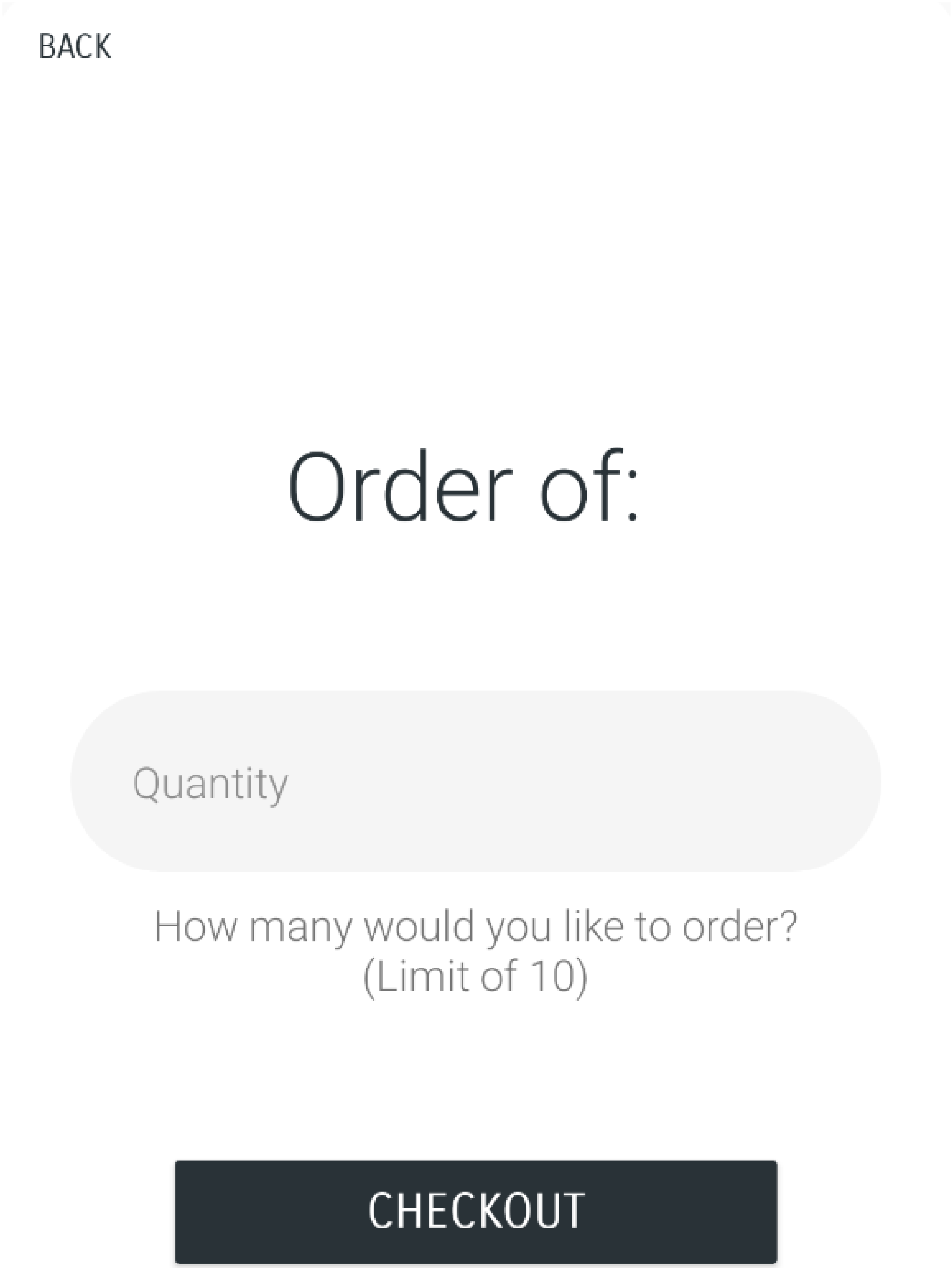
Item 5
Sub Item 5

Item 6
Sub Item 6


BACK
TextView
TextView
TextView
Item 1
Sub Item 1
Item 2
Sub Item 2
Item 3
Sub Item 3
Item 4
Sub Item 4
Item 5
Sub Item 5
Item 6
Sub Item 6

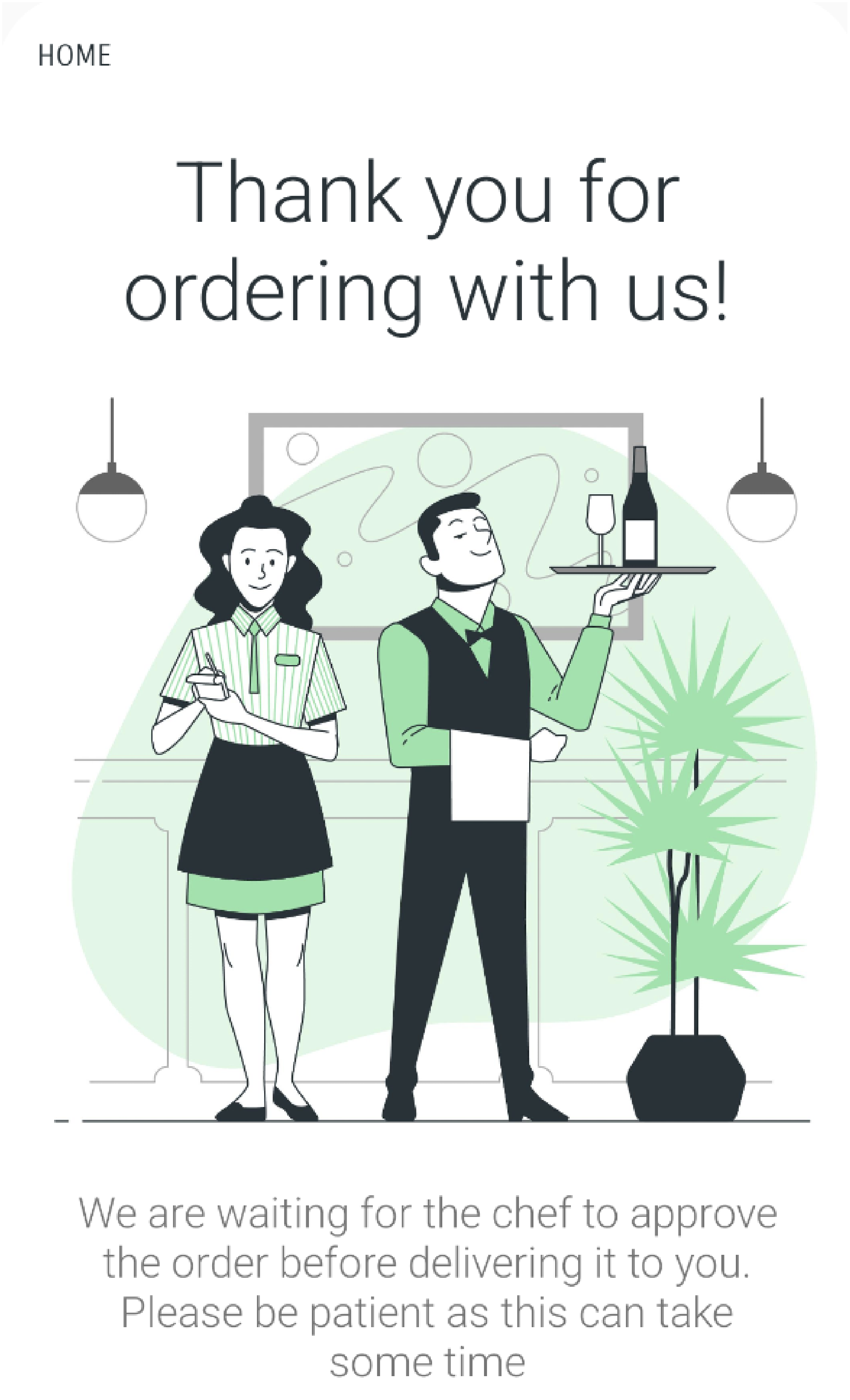


Click chef's name to see more



Order of:

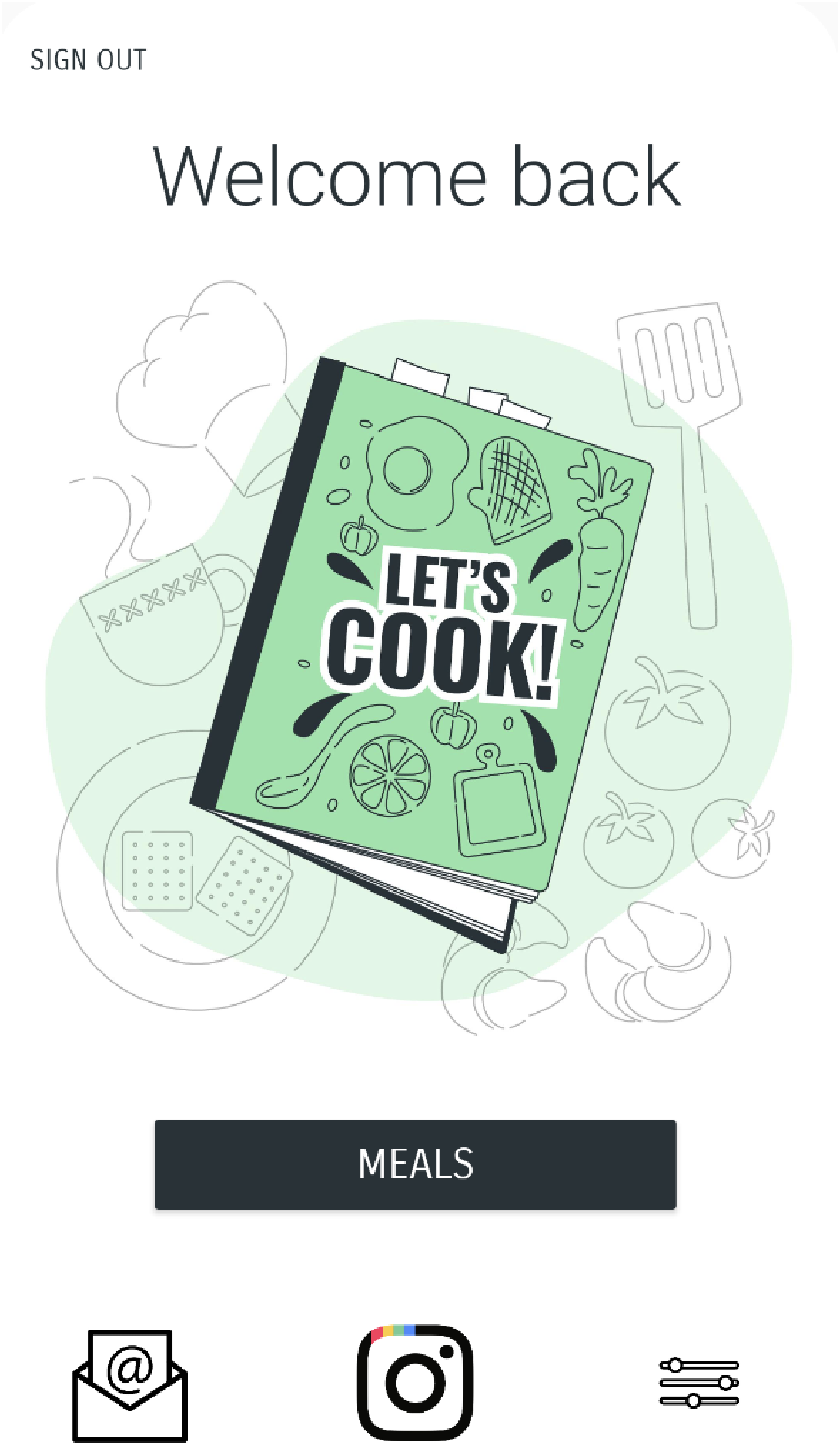
How many would you like to order?
(Limit of 10)



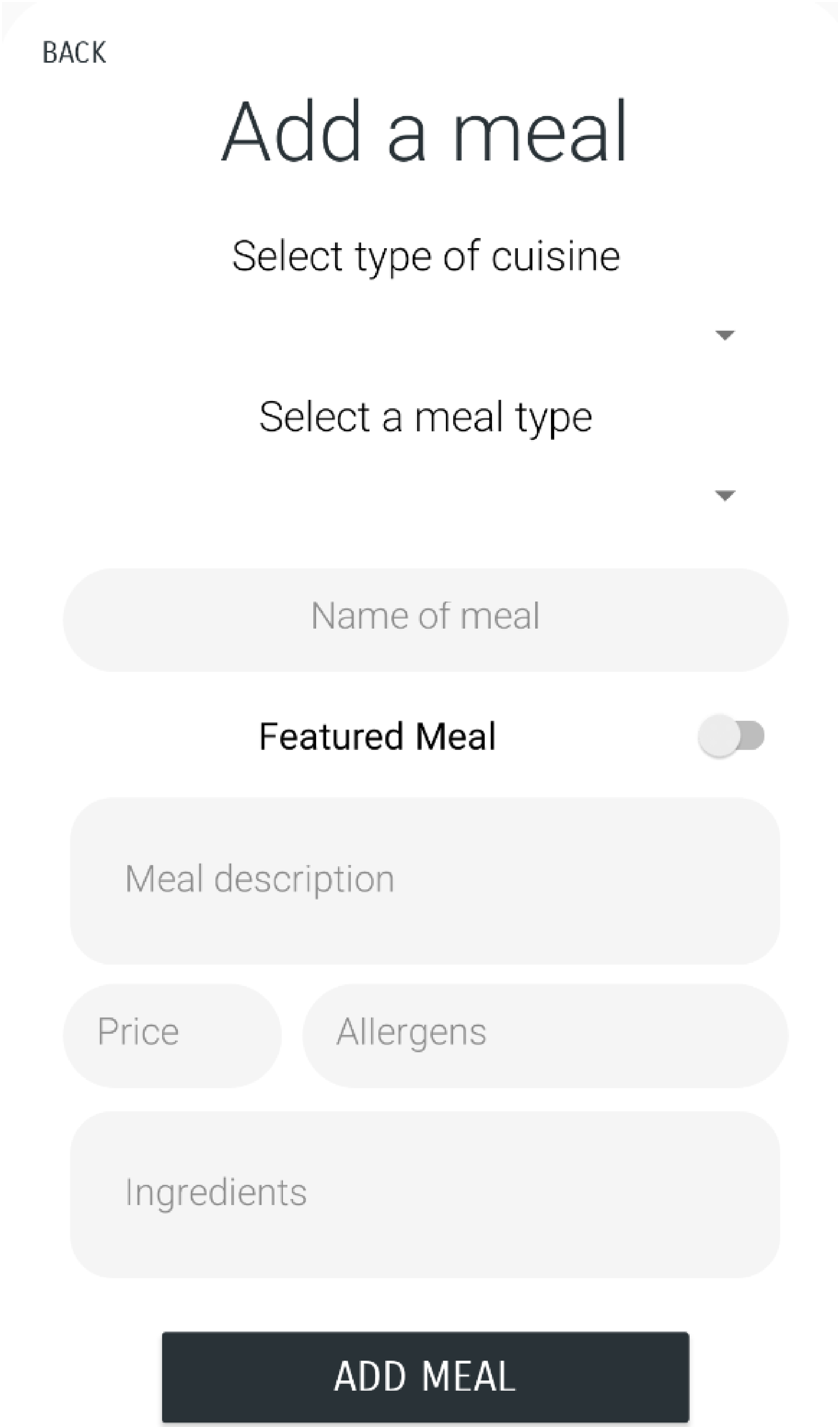
HOME

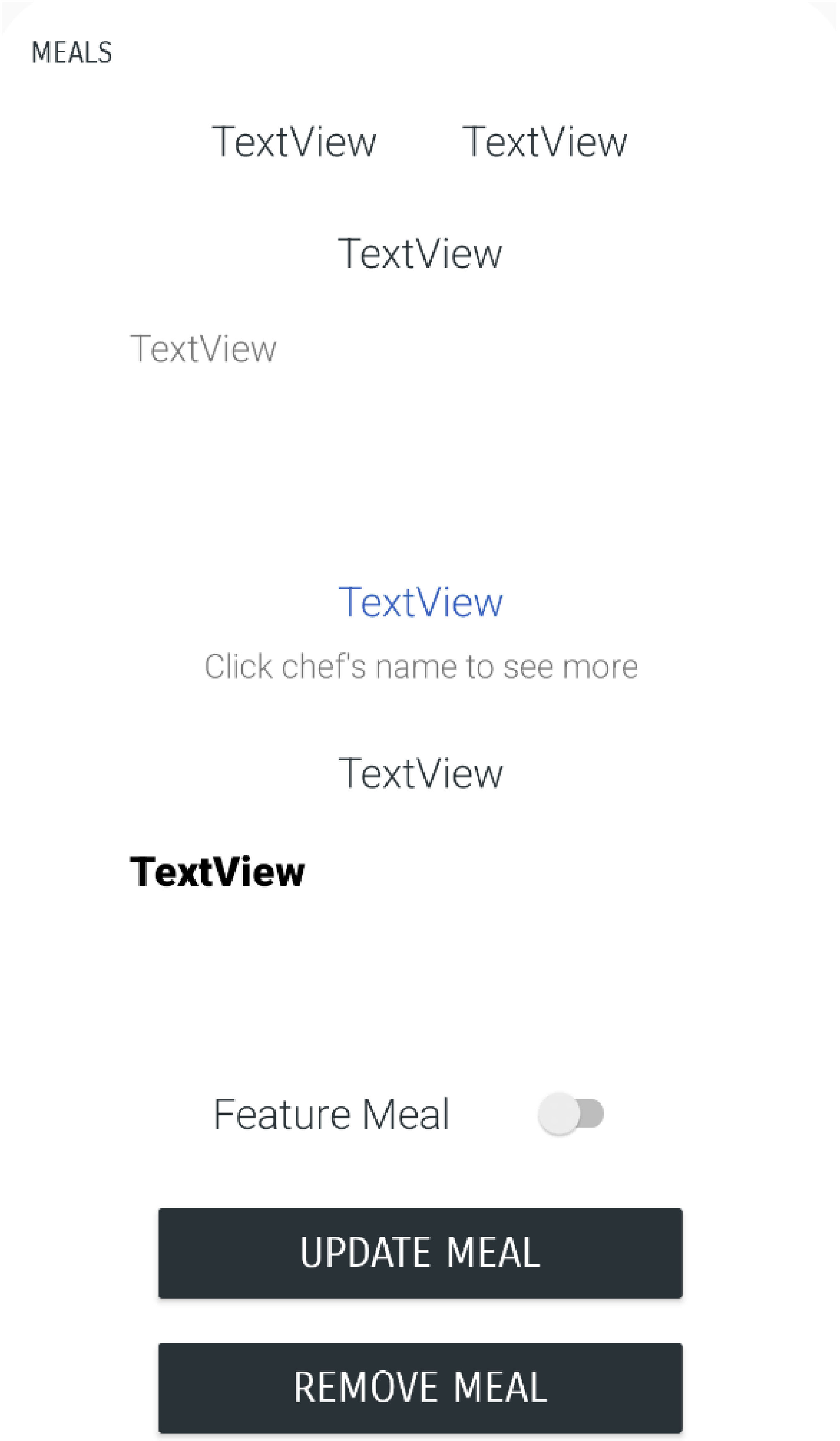
Thank you for ordering with us!

We are waiting for the chef to approve the order before delivering it to you. Please be patient as this can take some time

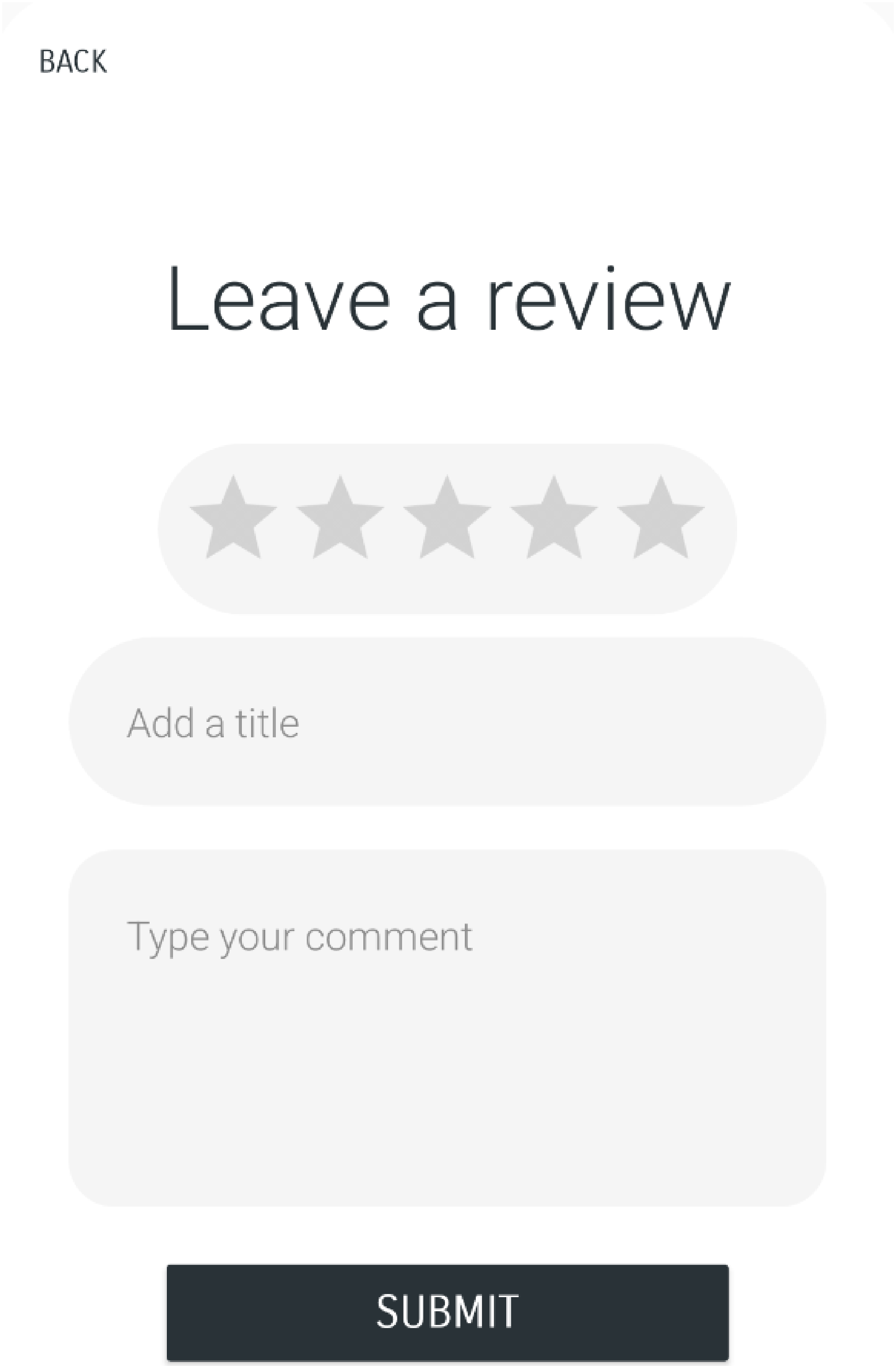


Welcome back


BACK
Add a meal
Select type of cuisine
Select a meal type
Name of meal
Featured Meal
Meal description
Price
Allergens
Ingredients
ADD MEAL



Click chef's name to see more


BACK
Leave a review
★★★★★
Add a title
Type your comment
SUBMIT

Lessons

We started our project on a foundation with high ambitions and started with a requirement analysis for deliverable 1. This habit of making a requirement analysis of every deliverable made our job much more complicated as we had to go back and update new features to accompany the necessary new requirements and lost us significantly more time than if we had a main blueprint implemented from all the deliverables combined. Some GUI lessons learned came from problems that would arise with text that has characters so long it can’t be displayed in an editText. A fix for this was understanding the importance of validators and making sure every editText has a validator not only to support non appropriate inputs but also for exception handling. Another main problem we faced was taking the easy way out in building most layouts using relative layouts. This was a huge detriment and setback. Learning to use Constraint layouts and linear/horizontal layouts helped us with using the app for any type of phone display. When you constraint things to a linear layout no matter the resolution of the screen it stays consistent in the display and understanding that was a huge step in the right direction for us in our user interface. Another important lesson we learned early is that following labs and tutorials won't get you far in building UI or implementation of the app past the bare minimum. We learned that if we want to excel beyond and be in the upper echelon we have to take the initiative to research videos and search up the majority of the features and ideas we wanted to implement. Moreover we decided to take the SQLite direction for our database needs and while it was harder in the short run trying to understand the code and implementation it is a stronger skill to have in your locker than FireBase which we noticed an overwhelming majority of groups gravitated towards. The biggest lesson we learned and what kept us focused was building our chemistry together as a group and making sure every person has assigned roles and everyone is held to a high standard and with accountability. Knowing that we are a group of three we knew we would have to work that much harder to beat all the other four member groups. So we did just that by having communication as our number one priority.