**topics to cover:**

1. git convention + gitHub desktop + postman
2. env file + constants + apiConfig file
3. virtual dom
4. routing
5. project structure and components splitting
6. parent and child component communication
7. understanding the package.json
8. webpack
9. styles with makeStyles
10. async await vs promise
11. js eventLoop
12. essential functions: map, filter, reduce, forEach, every and some vs for
13. middlewares (interceptors, thunk and googleAnalytics)
14. deep dive into interceptors
15. redux toolkit
16. localStorage
17. fields validation with formik
18. unit test with jest
19. http requests (using axios) + Promise.all
20. lazy loading
21. pagination

**installations**:

1. git - <https://git-scm.com/downloads>
2. node - <https://nodejs.org/en/download/>
3. gitHub desktop- <https://desktop.github.com/>
4. postman - <https://www.postman.com/downloads/>
5. vsCode

**test yourself:**

1. git --version
2. node -v
3. npm -v

**get started:**

1. we will create a react app using typescript
2. the app will have 2 routes:

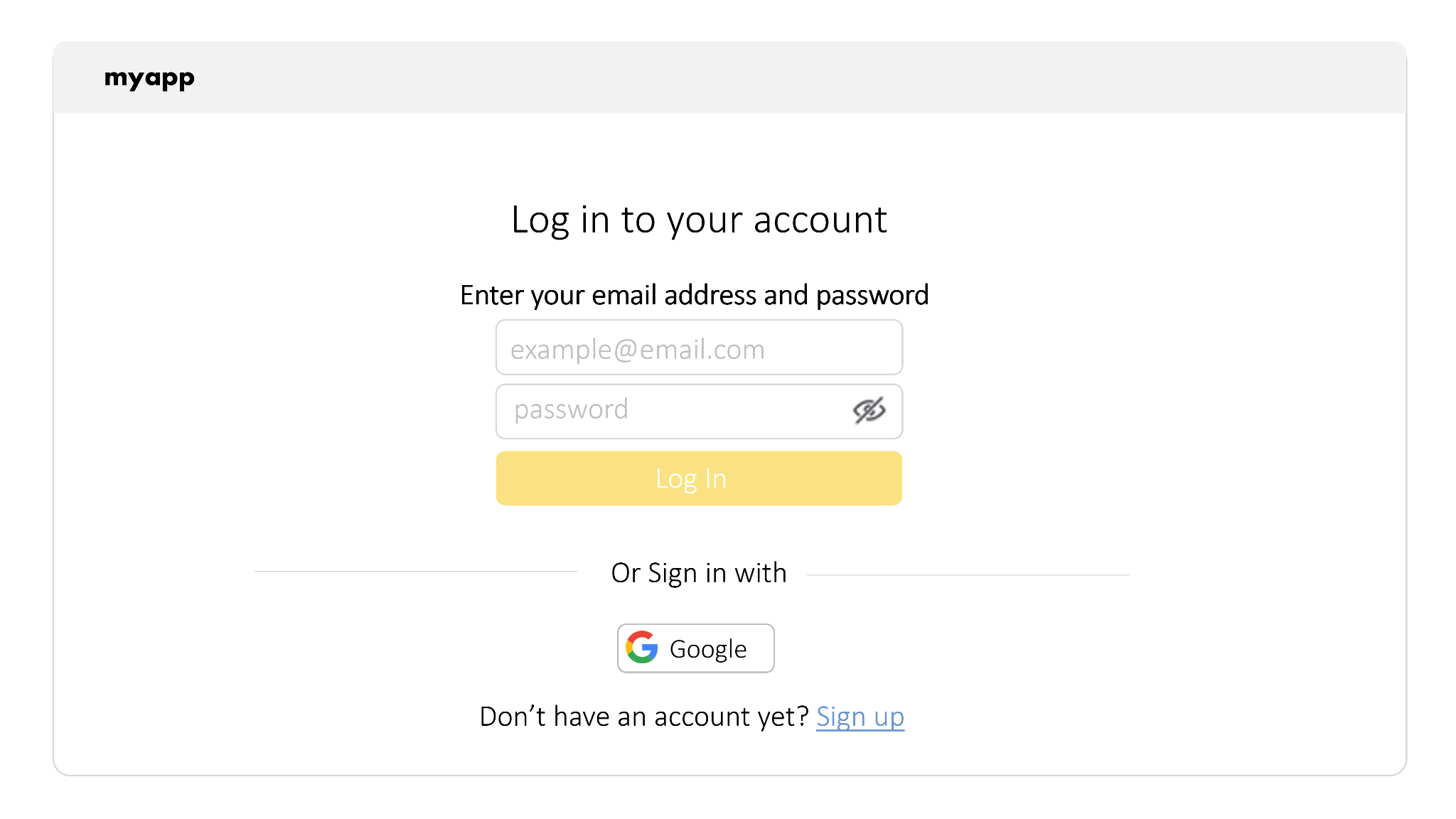
* login
* landing page

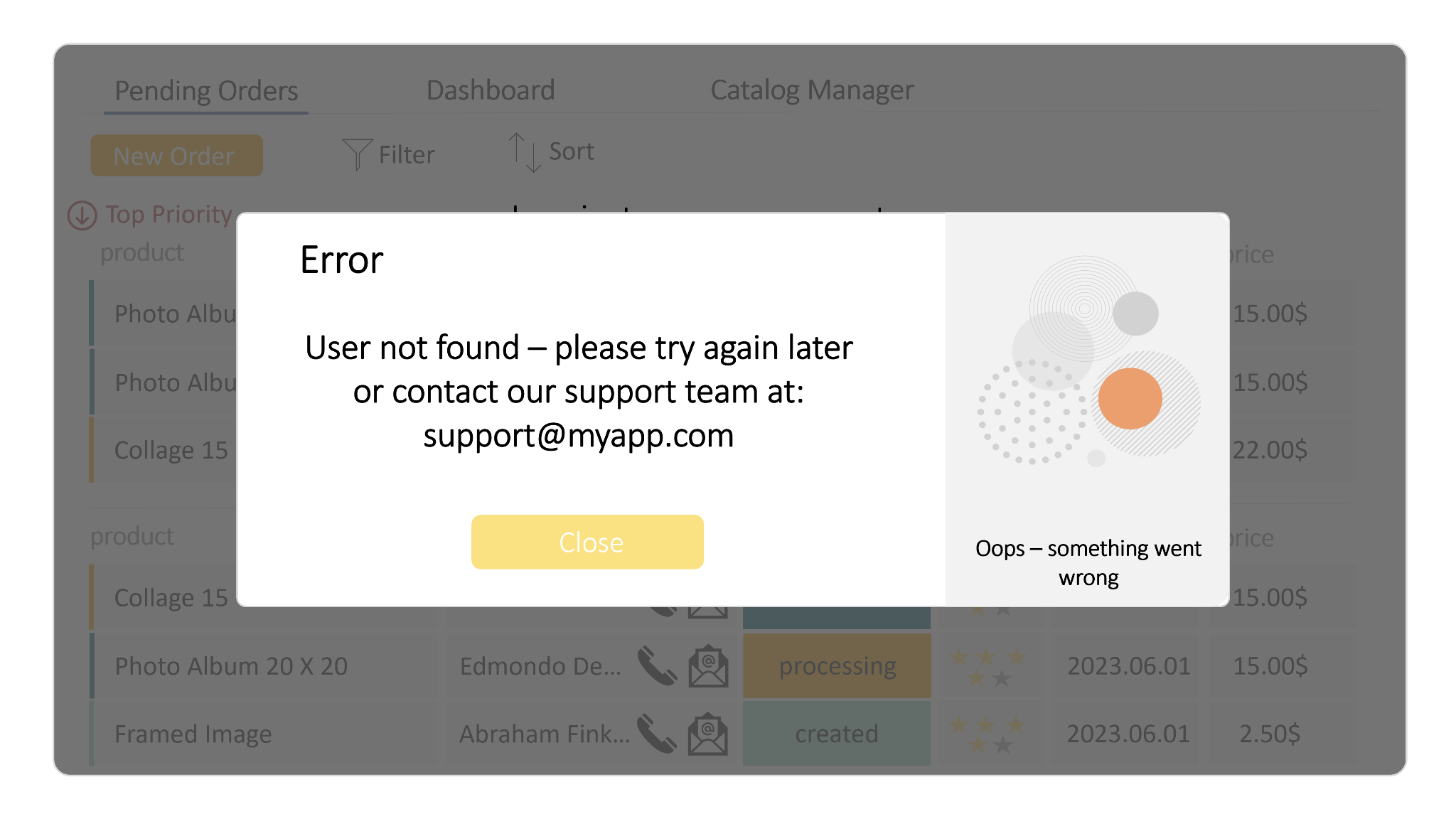
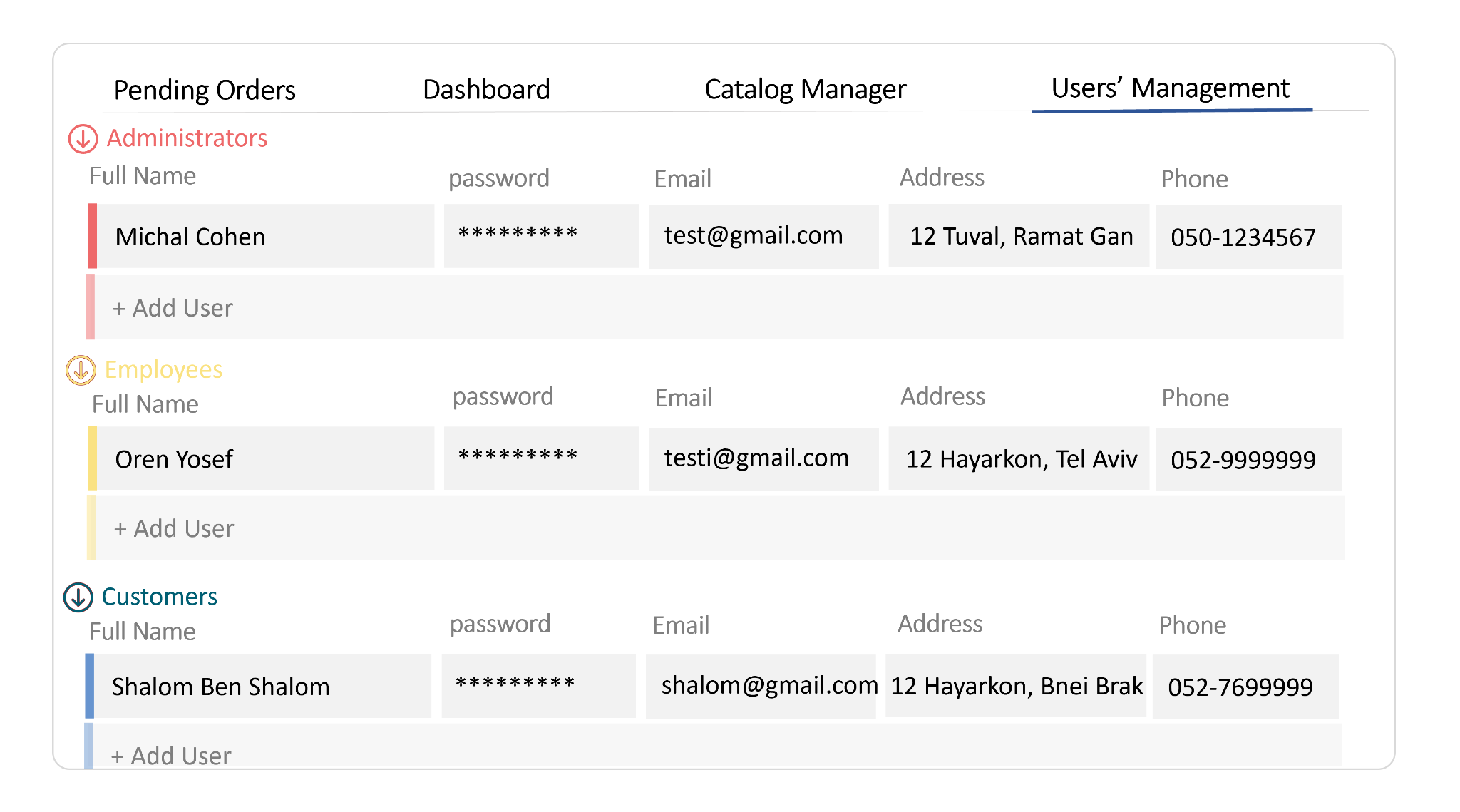
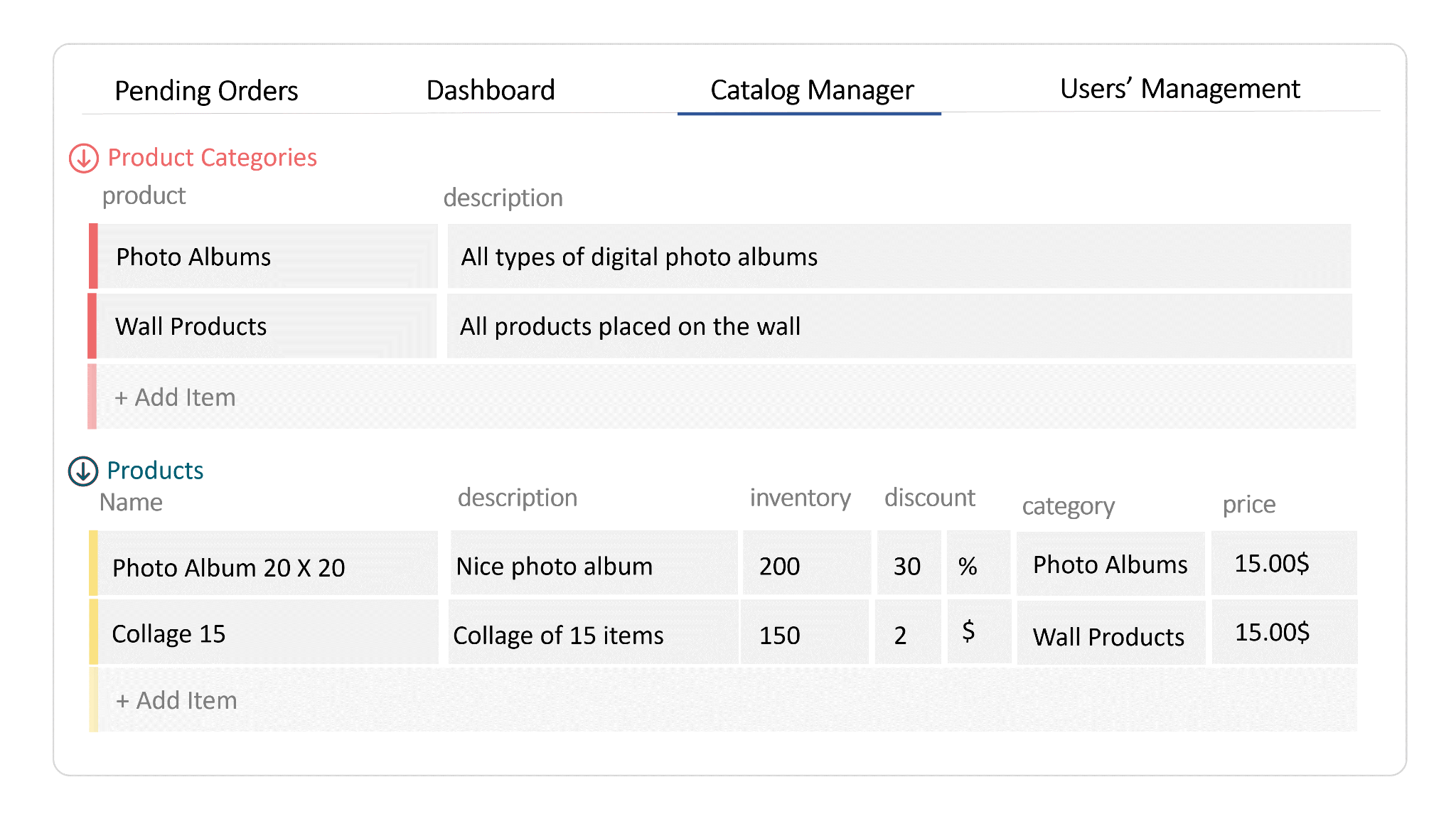
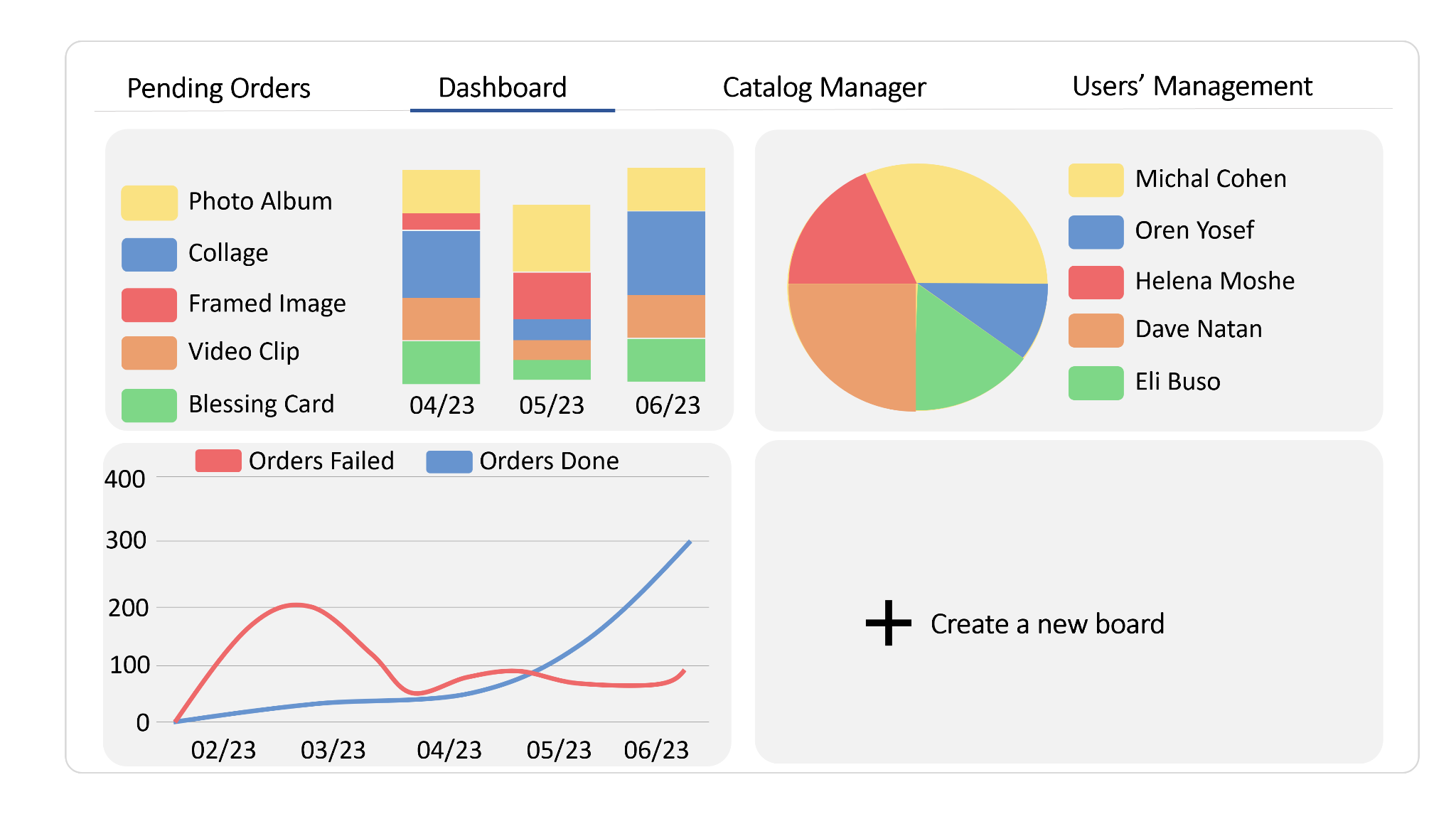
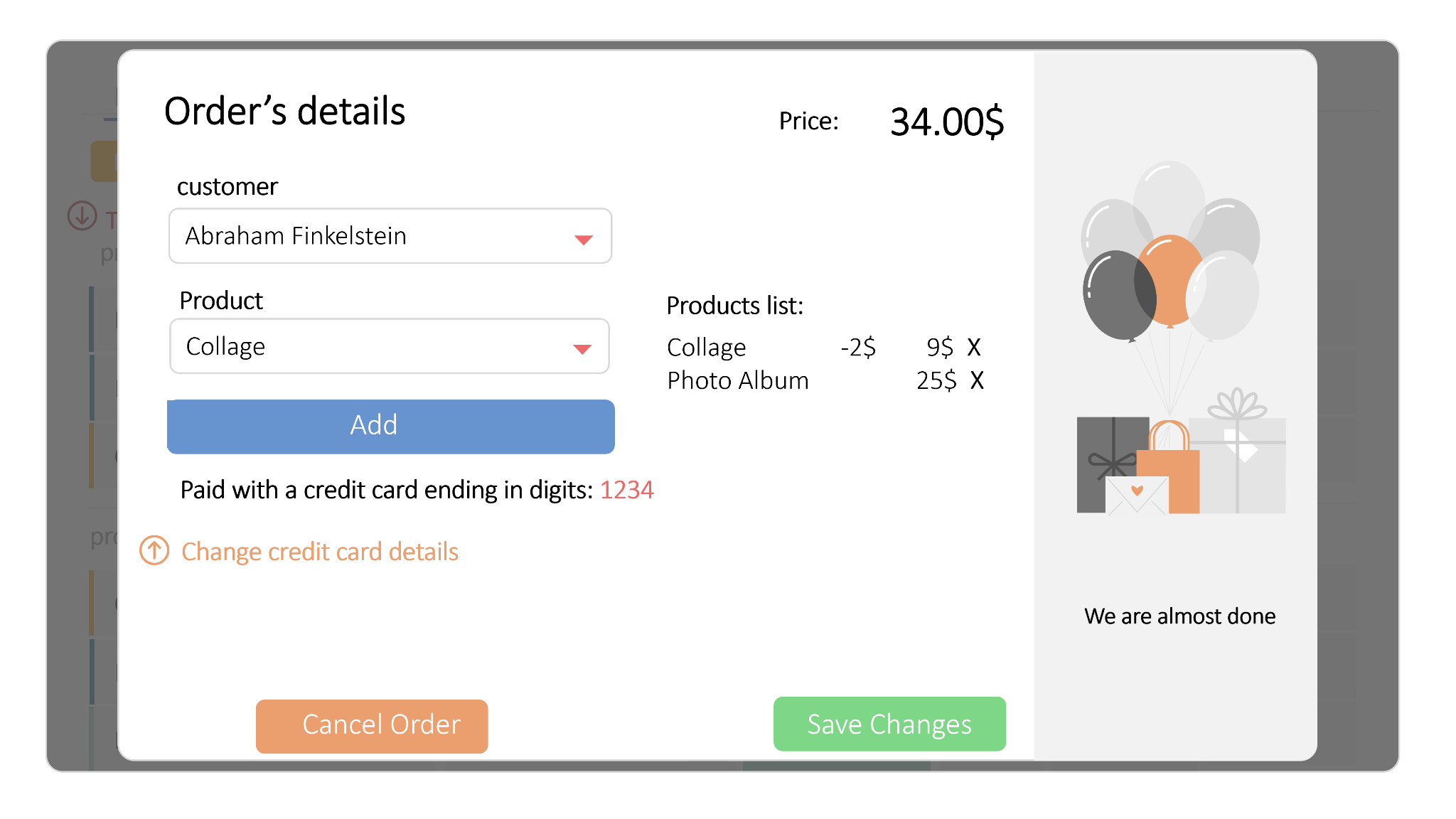
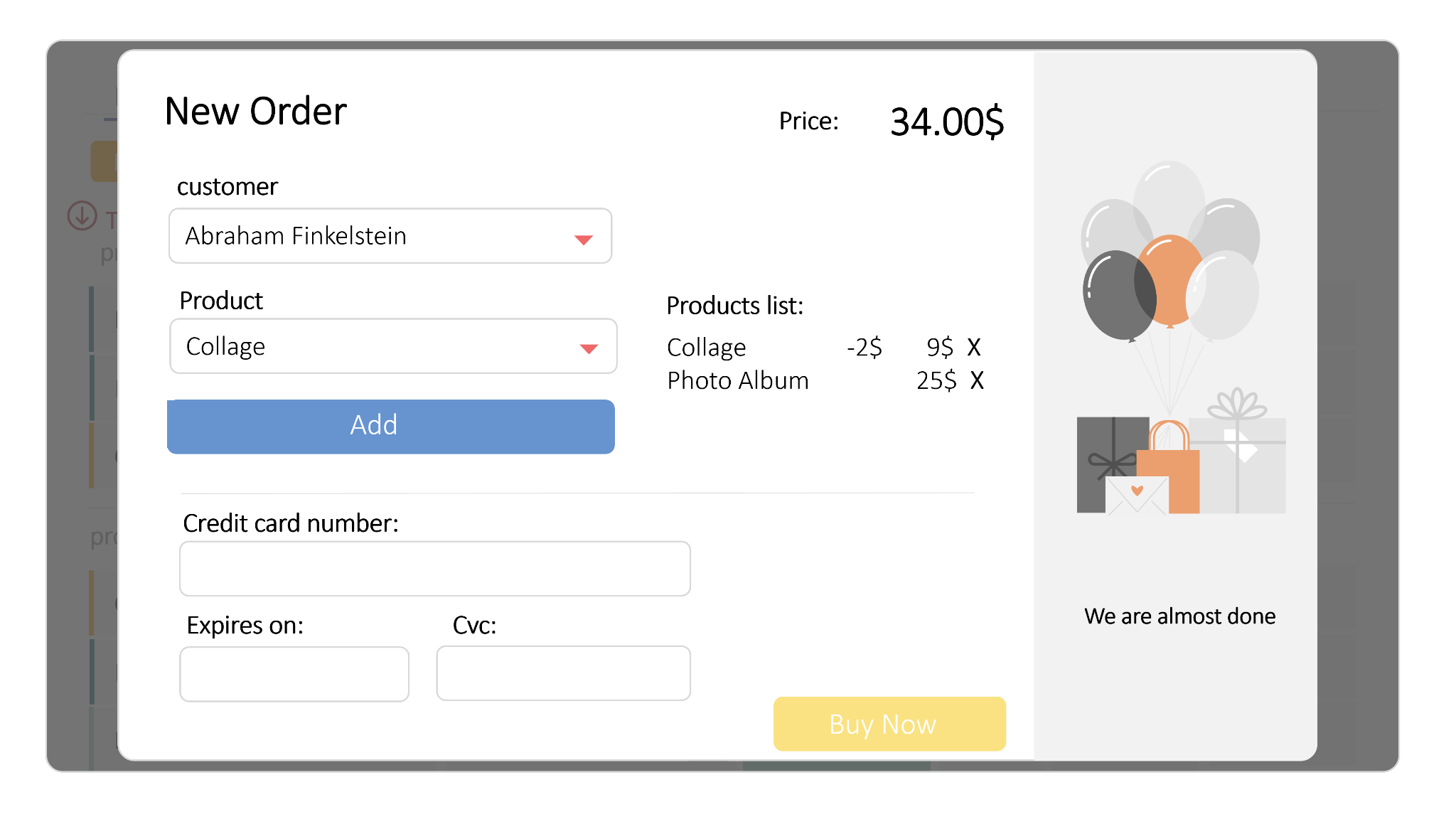
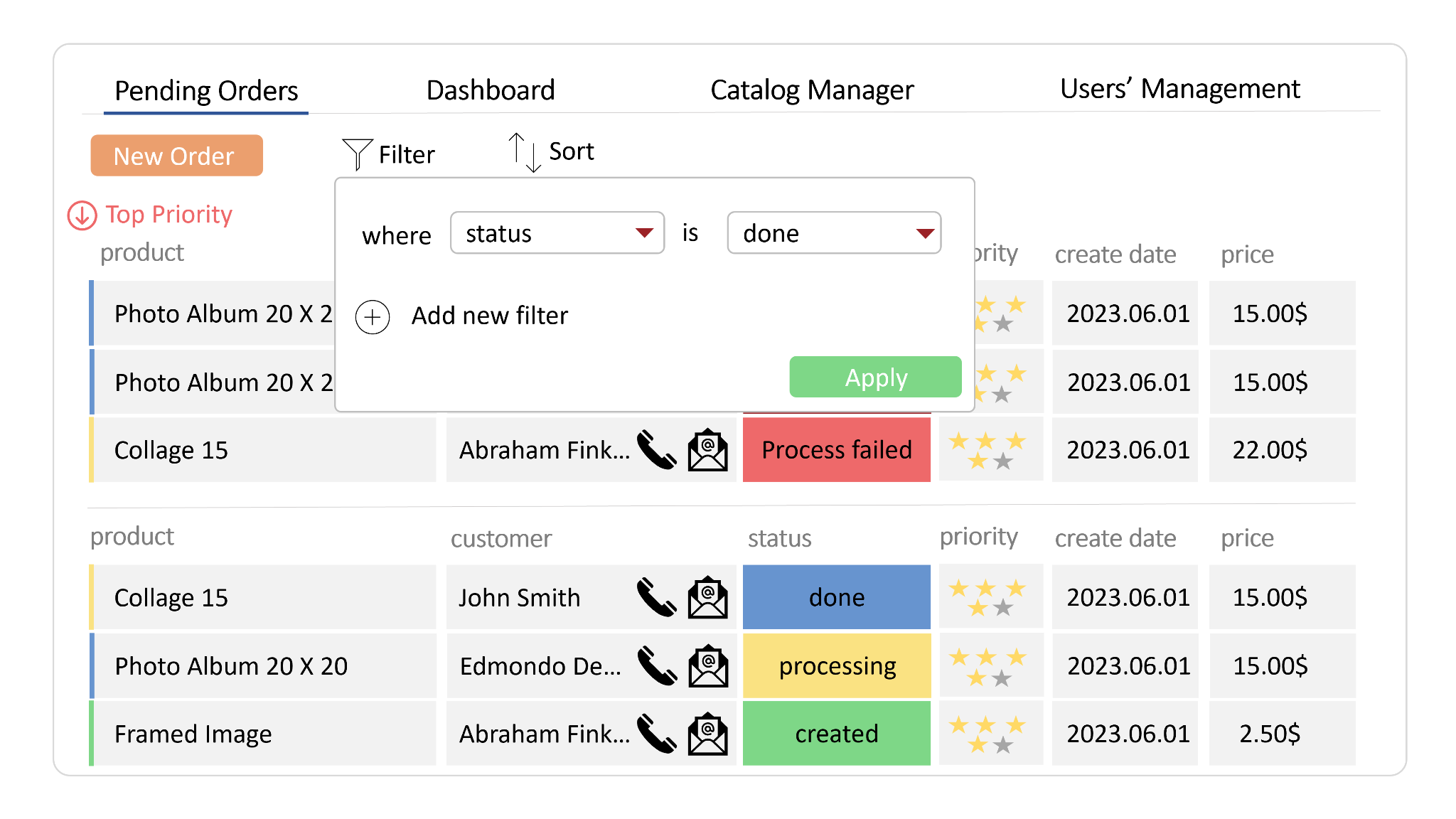
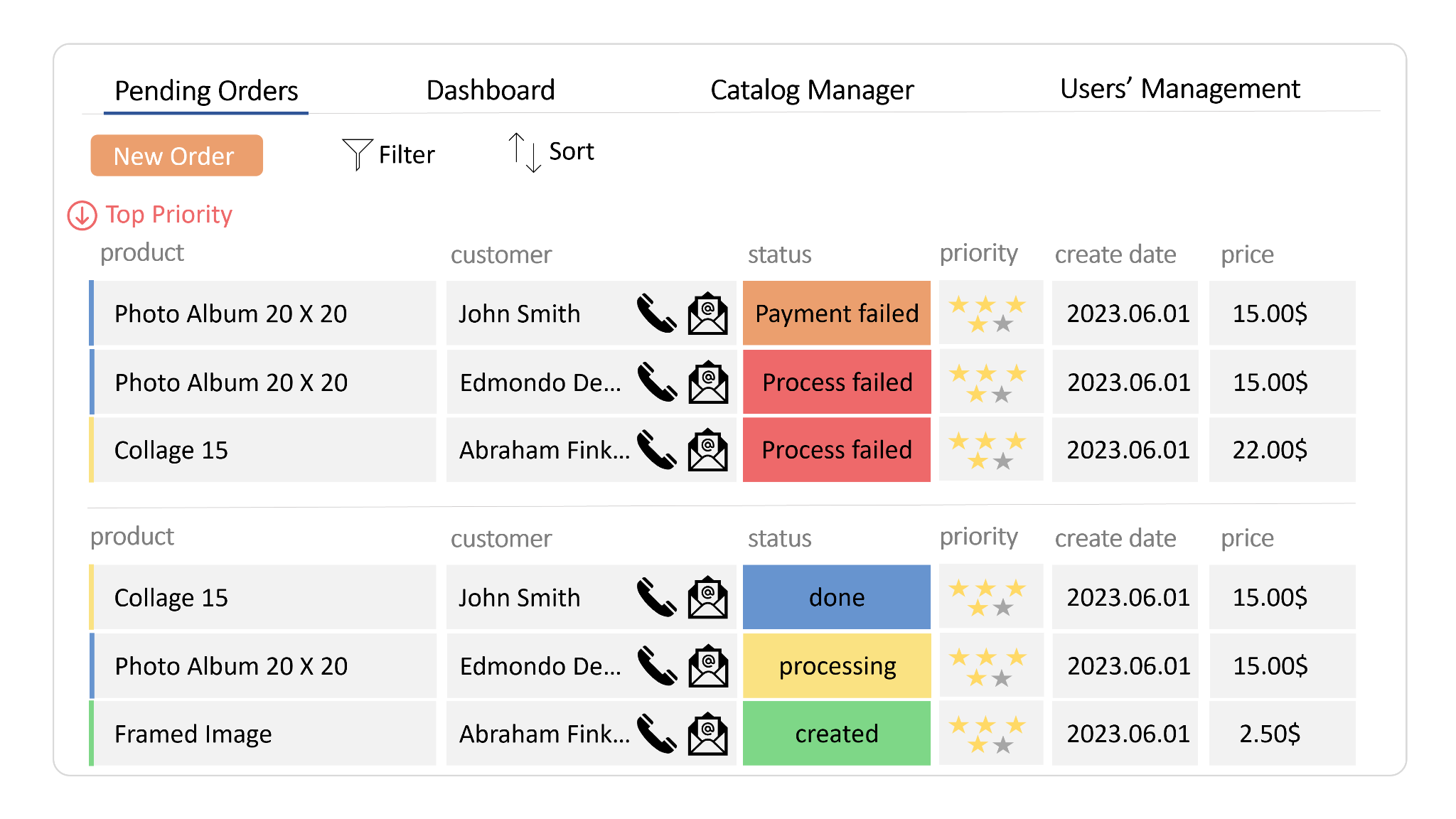
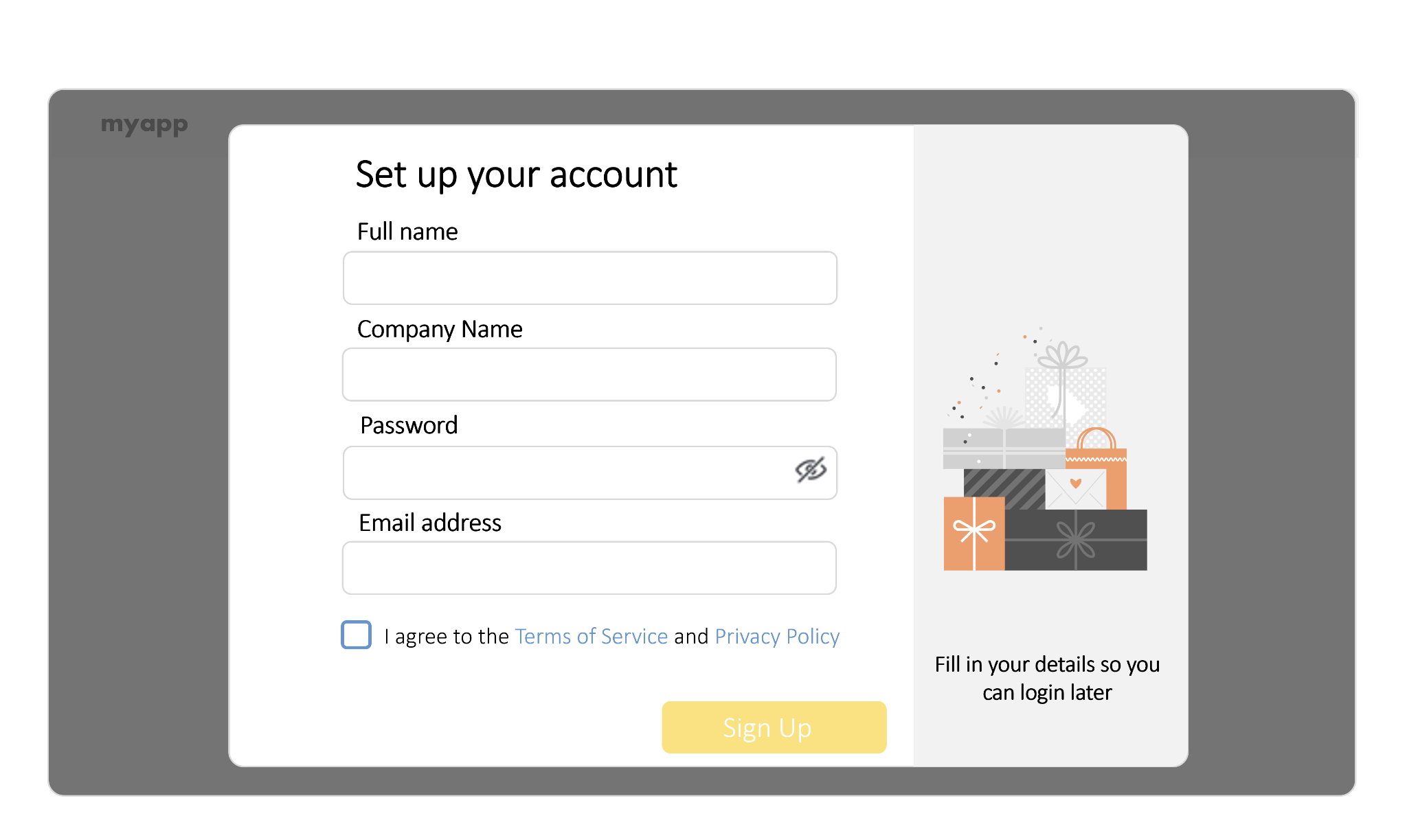
1. the login page will lead to a modal for signup for creating new account
2. the landing page will have 4 tabs:

* pending orders - display all the orders that are not completed yet, this page includes 2 tables, one for orders with top priority and the other one for the rest

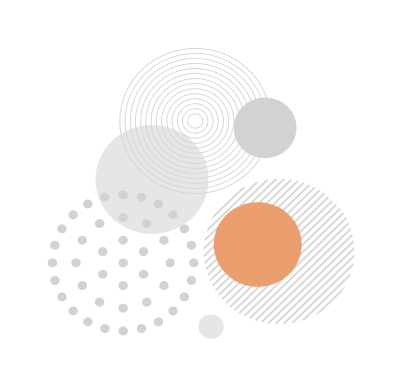
in the landing page employee can filter and sort the orders + create a new order + edit existing orders

* dashboard - will display 3 different graphs all of them will show some extra data about the system (top 5 employees, top 5 selling products and count of successful and unsuccessful orders)
* catalog manager (tab available only for admin users) - display all the products in the system and their categories - admin user can manage the catalog
* user’s management - for adding, editing and deleting users, employees can only add or edit customers, admin will have access for all the actions for all the users types

screens:



assets:



colors:

const PALLETE = {

BLUE: '#6794CF',

YELLOW: '#FAE282',

RED: '#EE696A',

GREEN: '#7ED787',

ORANGE: '#EB9F6E',

WHITE: '#FFFFFF',

}

tasks for sap:

1. implement google analytics as middleware
2. implement global error modal for all unexpected errors (use interceptors)
3. implement socket.io for every new order - whenever a new order is added by one user - all the other active users whose active screen is the "pending orders" tab - should see the new order added to their screen (without refresh)
4. phone and email interface - when a user clicks the phone icone or the email icon a caller or an email sender will open populate with the email or the phone of the customer