The project consists of a system that takes 4 parties into consideration:

– Me (who can create/delete dashboards)
– Restaurant owners (who can change/personalize options in the dashboard)
– Employees (who can use tablets to keep track of all the tables)
– Customers (who use the app to order items)

The project is ALL OR NOTHING. That means we do not accept half deliveries. This is to avoid people working on the easy parts and leaving the hard parts of the projects. In this case it cost us more budget and time to find someone to continue with the work. There's no guarentee for us we can find someone who is qualified to do so, nor is there any guarentee for us this said person can work with your code. So, either all points are finished, or the project will be cancelled.

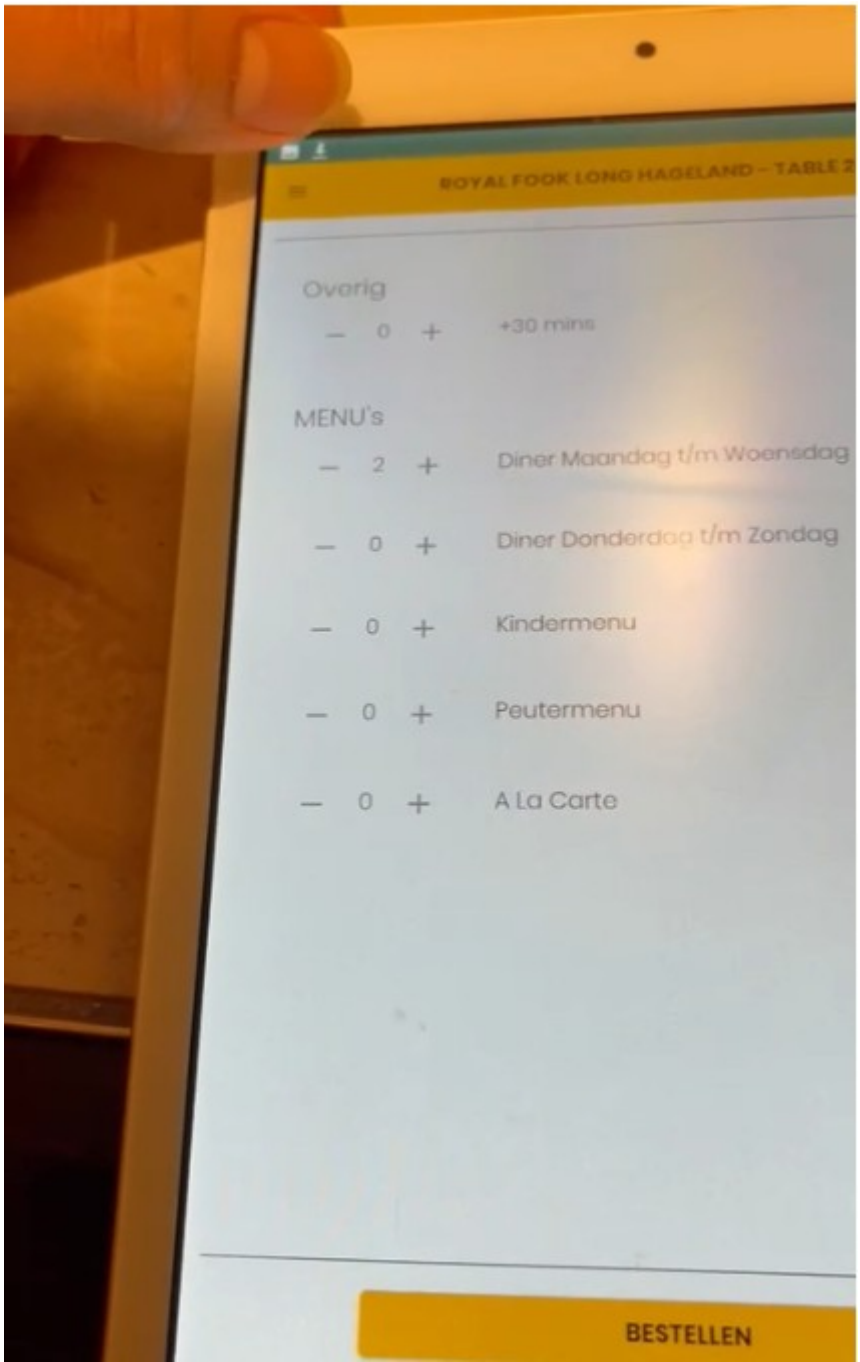
Customer side of things:

– Customer sit down on a table and can either scan a QR or go to the website and enter a code (that is matching the table they are sitting on).
– They have 2.5 hours to order items. Each person can order 5 items per round. So a group of 3 people can order 15 items total per round.
– There are unlimited rounds. Customers can see the history of their orders in the web app. They can also see the current bill.
– Search function for a dishes/items
– The items is a long list. The category bar can be used to jump to the right category. If you scroll down the list, the category bar will center the current item category.
– The categories and pictures of the dishes can be defined on the dashboard.
– Items might have specific information (vegan/gluten free etc.), can be added through the dashboard.
– Multiple people can add items until it reaches the limit, then one of the customers can press to send it.
– Customers can click on "Service" and then select to ask for a waiter, or whether he wants more wasabi, soy sauce, or giner. He can also ask for the bill.
– We are still deciding whether we want to implement payment. If we do, then they can select a tip and pay themselves and employees can see on tablets that they paid. Otherwise we'll need a button for calling for the bill. In that case, employees need to be able to mark the table as "paid" and then close it to record to the dashboard.
– All customers on the same table can see in real time what other people has ordered so far. So if one customer in the group adds one item. The counter will be 1/15 for all people around the table.
– After one of the people of the group presses to order, there is a 30 second window that people can cancel. If it's not cancelled, the order will be processed, and a receipt will come from the receipt printer.
– A popup when there is 30min left for them to order.
– For reference, check the images I send before. The screens and texts on there are also requirements!

Tablet for employees:

– Employees can see in the tablet which tables are available. They bring the customer to a table that's available and open the table. Before they open it, customers cannot use the QR or link.

– They must be able to select how many people and what kind of menu at the very start to open the table. There are different “all-you-can-eat” menu’s (for children, for different days, as Monday to Wednesday are cheaper). When table is opened this way, the customers can see on the web app how much it costs. Food is all-you-can-eat, but some items still cost money (and drinks also cost money) so it must add up correctly for the bill. Example on next page.



– An overview of all tables and icons to show whether they've paid, whether they service, or whether they need more wasabi, soy sauce, or ginger.
– Able to add extra time for customers if they want to.
– Able to close tables.
– If employees make a mistake, for example, they accidently pressed the wrong menu, or they added too many people, this must be easy to correct!
– Maybe a page with a map of the tables, but that's just an image we'll provide.
– Employees know what to do since every time customers order something, a receipt will be printed out.

Printers:

– These are receipt printers. They need specific format for it to look good. We can test this together. A receipt needs to come out for different cases: When customer needs a service, when they order something, when they pay etc. Also, we must be able to select which categories come out at which printer. The sushi printer for example will be placed near the sushibar, while drinks are placed elsewhere. We want to be able to select in dashboard which categories come out where. The formats are just slightly different. So whenever a receipt is printed, employeed know they need to do something.
– On the receipt you can see the dishes that are ordered and the table number.
– Dishes may have additional information, that is all showed in this receipt.

Dashboard:

– History of all orders with time and amount they paid etc. (standard statistics to keep)
– The customer web app has stuff that we need to be able to change or add here, such as: company logo, product categories, products (and their images and other information such as which categories they belong), chinese name of the product (as sometime this need to be printed on the receipt).
– Add and remove tables and/or QR codes.
– Change the code for a table link, so people can't make a picture of the QR and access the link from home. (www.whatever.com/restaurant-1/table?<code>
– Printers, choose which printer prints which category of items, and for services, and for the bill.
– Super user (probably me, since I handle stuff for my uncle) can select different dashboards, while employees can only see the dashboard in the restaurant they are working. So if my uncle or my

mom opens a second restaurant, I can have a overview of both. There must be an easy way for me to add dashboards. I think it's best if both restaurant works under same domain, with different extensions. www.whatever.nl/my_uncles_first_restaurant & www.whatever.nl/my_uncles_second_restaurant . So in the dashboard I can also decide on this extension. P.s. it must be on our domain of course.

Last things:
– Obviously we need to continue until everything is bug free.
– It must be secure. Think about things such as, the employees cannot suddenly access information from the other restaurants if they are not super user.
– Everyhing must be custom. **NO THIRD PARTY** stuff.