University of Victoria

Requirements Specifications (Client's Feedback)

Team Number:	1
Project Name:	ClassDASH
Analysts:	Zikai Hao, Pengfei Li, Shivani Ram
Clients:	Coby Lam, Emile Keruzore

Legend:
Requirement Selected
Requirement Discarded

I. Functional Requirements		M	S	C	W	Feedback/Observations
Food Orders						
1.1	All customers can order food via the mobile application	X				This should be a must as the main priority is a mobile application
1.2	Customers and workers should be able to have a browsing section with all the foods available and time restrictions on the menu	X				All user types (customers and workers will be able to browse menus)
1.3	Customers must be able to use a search bar to find specific items	X				

I. Functional Requirements		M	S	C	W	Feedback/Observations
1.4	Customers must be able to view recommended orders based off of prior orders	X				
1.5	Customers can leave a note in regards to any food allergies, preferences or customize options for a selected item(eg: burger, no cheese, no lettuce, extra sauce)	X				Needed for food allergies? May need to be a Must
1.6	The Customer will be able to review their order before confirming	X				
Payment Related						
2.1	All Customers can pay for their food using their ONECard, credit card, or debit card	X				
2.2	Users will create a system account to log in/manage their account and payments		X			All users regardless of payment methods log in: via their UVic account or create a guest account
Application Functions						
3.1	The application supports Customers' friends picking up their order				X	Moved to a Won't as this feature presents too much risk and edge cases
3.2	The application provides an embedded map function to help users get to the pick-up location.				X	Not necessary - could link to existing maps
3.3	An order through the app will have a unique identifier to ensure the right Customer receives their order (order ID)	X				
3.4	The application should mark an order as having been processed and remove it from the list of active orders		X			
3.5	The application should actively update: presenting up-to-date menus to customers and retrieving new orders from the database for workers		X			

I. Functional Requirements		M	S	C	W	Feedback/Observations
3.6	The application should alert Workers when a packed order has been sitting too long and poses a risk		X			
Food Worker-side Functions						
4.1	The system will have a food service interface that displays the orders (and the order ID) in a queue for Workers	X				How will the food-service-side update menus? Or mark items as sold out -- or even if not enough workers to process orders for a given vendor (Solved)
4.2	Workers must be able to create service-side accounts to manage their menus (mark items sold out/new items/new descriptions or prices)	X				
4.3	Workers can cancel the food order via the worker-facing side of the application if the food is not yet under preparation	X				
4.4	The app should pass basic security tests when transmitting information to and from application and databases		X			

II. Non-Functional Requirements		M	S	C	W	Feedback/Observations
Standards						
1.1	The application should have a minimum buffering time to ensure smooth user interactions		X			Minimal loading rather than buffering

II. Non-Functional Requirements		M	S	C	W	Feedback/Observations
1.2	The application must be clean and simple looking design for easy navigation	X				
1.3	The system should offer a faster ordering process than standing in line at peak hours		X			
1.4	The system's Customer- and Worker- facing sides should both be responsive and perform at similar speeds.		X			Irrelevant as they perform different tasks
Safety Related						
2.1	The system should encrypted users' personal information and financial transactions in ASCII standard		X			
Format Related						
3.1	The application interfaces only support English and French			X		We are just supporting English as that is the primary language spoken on campus
3.2	The System produces standard electronic receipts for users after order submissions			X		
3.3	The food pick-up confirmation should based on a QR-code system		X			QR code + ID? What will the process be to pick up the correct order → this will be a use case to map out
3.4	The format of time counting for food preparation will be a 24-hour system			X		Default setting will be 12-hour system however user may change this in the settings menu
3.5	All currencies in the payment processes must be calculated in Canadian dollar	X				

II. Non-Functional Requirements		M	S	C	W	Feedback/Observations
Accessibility						
4.1	The system must be accessible on both Android and iOS mobile systems	X				As our platform is a Mobile application, targeting both systems is essential
4.2	Application programming language should be python for maintenance purposes			X		Can be determined by developers/SMEs