Team No. 10

Team Members

- DeRon Sutton
- Aaron Gearhart
- Dawson Frick
- Chris Hardin
- Matt Chapman

Project Name

JayDash

Project Synopsis

This will be a mobile app to allow students to order food from KU Dining without waiting in line, from the convenience of their phone.

Project Description

• *Why is the project being undertaken?*

Our project is being undertaken to improve the efficiency of ordering food at KU Dining facilities. Every day during peak hours, such as lunchtime, students and faculty pack the DeBruce Center and The Underground, causing them to waste time in line. We want to create another way for students to order food and get in and out of high-traffic areas faster. This also has potential to help out faculty workers, because orders would come up on screen and they could just have them ready for pick up, and when you check out all you have to do is show some proof of purchase, which would reduce the amount of people that stand in line to pay.

• *Describe an opportunity or problem that the project is to address.*

Students forming into large lines which eventually overwhelms the dining staff and slows down overall efficiency. This will also allow for more social distancing in the case of a pandemic because there will be less people waiting for their food.

• *What will be the end result of the project?*

The end goal of our project is to design an app for both the students and the dining staff that will help dining operate more efficiently and prevent congestion within KU dining facilities.

Project Milestones

First Semester (Fall 2020)

1. Use Diagrams for Front and Back End Completed
2. Tasks established for each team member – With Descriptions
3. Video Presentation Completed

Second Semester (Spring 2021)

1. Use front end and back end diagrams to implement our design on a mobile device
2. Add functionality to our design and begin testing
3. Add connections between client side and dining side for seamless experience

• *Estimated completion date for each milestone.*

First semester (Fall 2020)

- A) Tasks established for each team member – With Descriptions – October 16th
- B) Use Diagrams for Front and Back End Completed - October 23rd
- C) Video Presentation Completed – October 26th

Second Semester (Spring 2021)

- A) Use front end and back end diagrams to implement our design on a mobile device (March 15th)
- B) Add functionality to our design and begin testing (April 15th)
- C) Add connections between client side and dining side for seamless experience (May 10th)

Project Budget

  • *Hardware, software, and/or computing resources*

    Software:

      XCode with Swift on MacOS

      Core Data – an iOS database manager

    Hardware:

      Cellphone/Tablet (iPhone/iPad, although we have these)

      3 Mac Books to work in iOS ecosystem

  • *Estimated cost*

  Mac Books: 3 units at $1299 per unit: $4000

  • *Vendor*

    Apple

  • *Special training*

    Learn Swift and Core Data or any other frameworks associated with Swift

  • *When they will be required*

    These will be required before next semester, so we have enough time to get familiar with Swift.

Preliminary Project Design

As stated above, the preliminary design of our app, JayDash, will consist of two different facets depending on if you are the student/faculty member making the order on an iPhone or if you are KU Dining taking and fulfilling the order on an iPad. Both combined will create a seamless ordering experience at KU's dining centers.

  Starting with the iPhone app, the user will be greeted to the following screen upon opening.
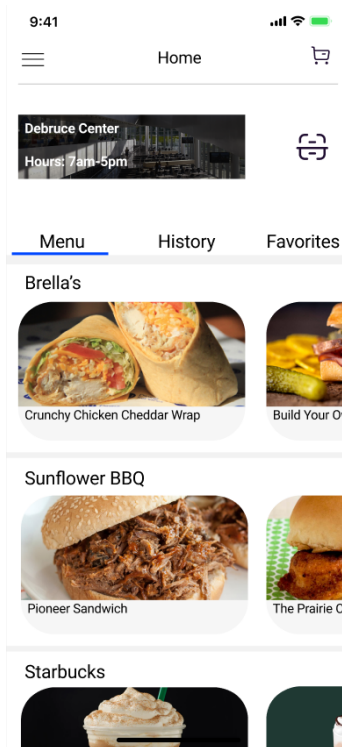
We wanted to keep the design of the iPhone app (as well as iPad app) as intuitive as possible so the user could order what they wanted as fast as possible with their busy schedules. Based on which dining location they select (which they can change by tapping the black box where it says *DeBruce Center*), the app will pull the hours, menu, prices from a database we will setup with said information. As seen by the mockup, the items are broken down by items in the different "shops" of the dining center, making it easy to find the food the user wants. There is also a *History* tab where the user can see past orders as well as a *Favorites* tab where they can quickly add their favorite items to their cart. Say the user selects the *Crunchy Chicken Cheddar Wrap*, they will be taken to a page that will allow them to select the toppings they want as well as what they do not want. Then they can add to their cart once they are done. If that is all they want, they can go to checkout by tapping the cart icon on the top right where they will be presented with a screen with what they ordered, the cost, and added tax/discounts, if applicable. In terms of making the payment, our plan is to implement Apple Pay initially, but then add other methods of payment later if leftover time. Once the user makes a payment, the order will be sent to the backend to be added to the user's past orders and to be sent to the iPad app so KU Dining can fulfill it. If we have time later next semester, we are planning on utilizing this information (the past orders) in conjunction with Swift's CoreML to predict what the student will order in the future and offer suggestions.

Below is a mockup of how the iPad app will look for KU Dining.
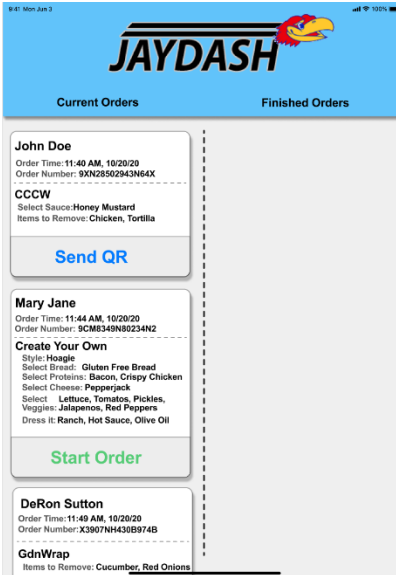
iPad App

When a student places an order, it will be sent to the dining side of the app, which shows two columns, current orders and finished orders. The current orders column will be a scrolable list of incoming orders. When an employee is ready to start the order, they will press the [Start Order] button which will alert the student that the order has started. After this is pressed, the button will then switch to a [Send QR] button. When the order is ready to be picked up, an employee can press this button, which will alert the student that their order is ready and for them to scan the QR code. This will eliminate an unnecessary paper trail of receipts during the transaction. After the QR code is sent, the order box moves over to the finished orders column and will display that it is ready for pickup. Once the order is picked up, it will automatically close the order and provide a start and pickup time and be stored. Below is a mockup of how the iPad app will look when the order is waiting for pickup and finished.
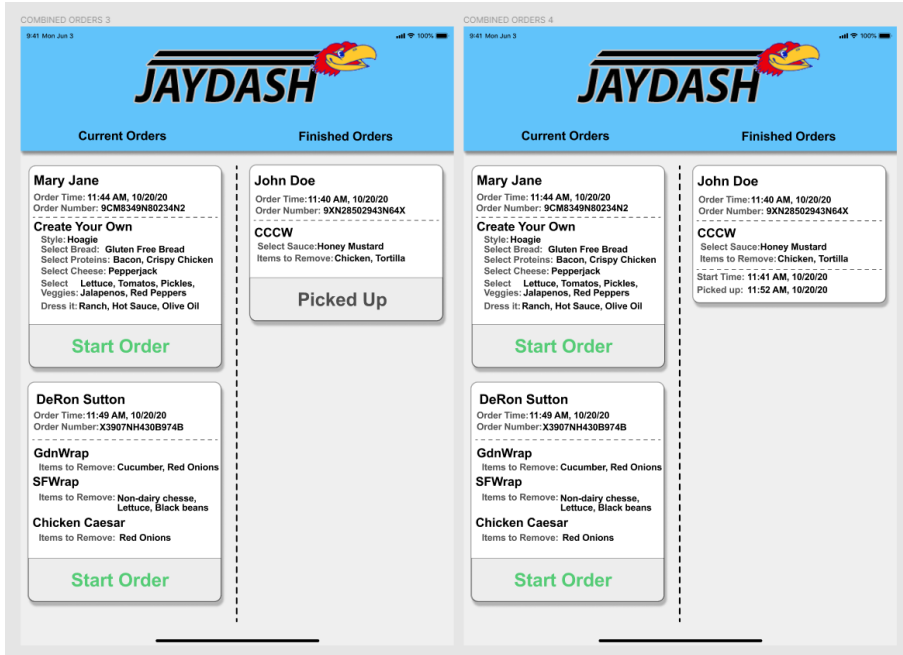
Databases

Our current design choice is to utilize one of Swift's included libraries called Core Data Cloud Kit. Core Data is the in-house database that swift offers and Cloud Kit enables us to allow customers on IOS to access the data from the app on all their iOS devices. Core data is useful in our project because it not only allows us to store complicated data structures locally but will also enable us to query and filter through that data which is not usually allowed when storing data in user-defaults. Below are some examples of tables we would expect to use in our database. These tables will serve as the backbone for user information, displaying menu items, order processing, and order history.

id	first_name	last_name	email	password
1000000	Alice	Jones	a123j234@ku.edu	$hashed$
1234567	Bob	Smith	b234s345@ku.edu	$hashed$
2222222	Jane	Doe	j345d456@ku.edu	$hashed$
9876543	John	Doe	j987d876@ku.edu	$hashed$

Users Table - stores login and id information for students and faculty.

location_id	restaurant_name	item_id	item_name	price
11	Chick-fil-A	1101	Chicken Sandwich	3.89
11	Chick-fil-A	1102	Spicy Chicken Sandwich	4.19
12	Brella's	1201	Crunchy Chicken Wrap	5.45
22	Brella's	2201	Crunchy Chicken Wrap	5.45
23	Sunflower BBQ Co.	2301	The Pioneer	6.25
23	Sunflower BBQ Co.	2302	The Prairie Chicken	6.25

Items Table - stores the menu items available at KU Dining Locations and their prices.

location_id	restaurant_name	location	food_court
11	Chick-fil-A	KU Memorial Union	0
12	Brella's	KU Memorial Union	1
22	Brella's	DeBruce Center	2
23	Sunflower BBQ Co.	DeBruce Center	2
1	FOOD COURT	KU Memorial Union	0
2	FOOD COURT	DeBruce Center	0

Locations Table - stores location of each restaurant and the id of the location. The food_court attribute gives the location_id to pick up the food or 0 if it is a pickup location.

order_id	pickup_location_id	user_id	total_cost	payment	ready_for_pickup
65	11	2222222	8.08	TRUE	TRUE
66	2	1000000	11.7	TRUE	TRUE
67	1	9876543	10.9	TRUE	FALSE
68	2	1234567	6.25	FALSE	FALSE

Orders Table – stores the order id as well as the order's total cost, who ordered it, where to pick it up, and its status.

order_id	item_id	order_location_id	quantity	user_id	comment
65	1101	11	1	2222222	no pickles
65	1102	12	1	2222222	no pickles
66	2201	22	1	1000000	
66	2301	23	1	1000000	
67	1201	12	2	9876543	add honey mustard
68	2401	24	1	1234567	side choice: spicy slaw

Contents Table – stores the items that were purchased in each order.

Design Constraints

One of our design constraints in this project is our decision to use Apple's XCode development environment across the board. This isn't a decision that was required to make this project a success, but this would help in having each part of the project (front-end, back-end, database, etc.) work better together because they are a part of the same environment. But this could limit us from taking advantage of some capabilities that other technologies may offer. This also prevents us from having cross platform support with Android devices, which limits the size of our potential userbase.

Ethical Issues

- Potentially having access to students' dining plan and/or credit card information
    - This could be a potential issue because this is an app that deals with transactions with money and this can be problematic if you set it up the wrong way. For our purposes right now, we are currently planning to implement Apple Pay as a method of payment. The user's iPhone will already have their credit card information, so all we would have to do is to invoke an Apple Pay API. For future iterations with adding KU's proprietary dining plan, we would have to take a couple of extra steps in ensuring that the user's account is protected in order to prevent students from losing their information and money.

Intellectual Property Issues

- Existing food delivery and order apps such as Grub Hub and Door Dash
    - There are already multiple existing applications to order food and have it delivered to you in app stores, which could make it difficult to avoid intellectual property issues. Creating our own food ordering application isn't exactly where the intellectual property issues will come up; instead the issues could come up based on where we get our design inspirations. It is imperative that our application doesn't steal any design applications, especially from apps like Grubhub and DoorDash. If we were to have a similar idea to Grubhub and DoorDash, and our design was also the same, it wouldn't be hard for those companies to say we stole their intellectual property and shut our app down.
- Ability to use KU logos and Copyrighted names such as Chick-fil-a and Starbucks.
    - There could also be intellectual property issues when it comes to displaying the KU logo within our app. For most institutions, whenever a 3rd party wants to use their logos; that 3rd party is usually required to receive permission from the institution to use their logo within their app. This is required to prevent 3rd parties from putting the institutions logo on something that could harm their brand. So, we will probably have to ask to utilize the KU logo. If we don't get permission from KU and our application doesn't align with the beliefs of KU, then KU could have our app removed from the app store and shut us down. The same reasoning can be applied to using franchise names such as Chick-fil-a and Starbucks and their logos or trademarked menu items.

Change Log

Switched to iOS

- We have changed the language/ecosystem we are going to be developing, going from using Flutter/Dart & ReactJS (cross-platform) to strictly iOS with Swift and Core Data.

Added 3 Mac Books to our budget

- To accommodate our change to iOS, we required Mac Books to complete our work in this ecosystem.

Team 10 Gantt Chart

Matt Chapman, Dawson Frick, Aaron Gearhart, Chris Hardin, DeRon Sutton

PROJECT TITLE	JayDash		COMPANY NAME	Team 10

TASK NUMBER	TASK TITLE	TASK Workers		9/21 M	T	W	R	F	9/28 M	T	W	R	F	10/5 M	T	W
Fall Semester																
1	Tasks established for each team member	All 5	5	▓	▓	▓										
2	Video Presentation	All 5	25													▓
3	Use Diagrams for Front End and Back End	All 5	12													
4	UI Design	Aaron, Chris	12													
Spring Semester																
1	Implement design on mobile device	All 5	55													
2	Add functionality to design	All 5	60													
3	Testing functionality	All 5	35													
4	Connect client and dining side	All 5	20													

SEMESTER ONE (FALL 2020)																																
		10/12					10/19					10/26					11/2					11/9					11/16					11/23	
R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W

	11/30						12/7						12/14						2/1						2/8						2/15						2/22			
R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R

		SEMESTER TWO (SPRING 2021)																																	
	2/29					3/7					3/14					3/21					3/28					4/4					4/11						
F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F

	4/18					4/25					5/2					5/9				
	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F