

**ALMA MATER STUDIORUM - UNIVERSITA' DI
BOLOGNA**

SEDE DI CESENA

FACOLTA' DI SCIENZE MATEMATICHE, FISICHE E
NATURALI

**CORSO DI LAUREA IN SCIENZE E TECNOLOGIE
INFORMATICHE**

**APPLICAZIONE SMARTPHONE
PER GESTIONE ORDINI DA REMOTO**

Relazione finale in

MOBILE WEB DESIGN

Relatore

Dott. Mirko Ravaioli

Presentata da

Giorgi Stefano

II SESSIONE di Laurea

Anno Accademico 2011/2012

Sommario

1	INTRODUZIONE	4
1.1	Smartphone	4
1.2	Sistemi Operativi	7
1.3	Vantaggi e Svantaggi Sistemi Operativi	9
1.4	Mercato Sistemi Operativi	11
1.5	App Store	12
2	DESCRIZIONE DEL PROGETTO	14
2.1	Descrizione Generale	14
2.1.1	Visualizzazione dei locali	15
2.1.2	Gestione dell'ordine	18
2.1.3	Gestione Ordini del Cameriere	20
2.2	Strutture Dati	21
2.3	Le classi del progetto	23
2.3.1	Categoria	24
2.3.2	Personalizzazione	24
2.3.3	Valore	25
2.3.4	Portata	25
2.3.5	Porzione	26
2.3.6	Comanda	26
2.4	Il Server	27
2.5	Diagramma di flusso Ordine del Cliente	28
2.6	Diagramma di flusso Gestione Ordine Cameriere	29
3	IMPLEMENTAZIONE DEL PROGETTO	30
3.1	Scelta dell'applicazione : Nativa o Web	30
3.2	Scelta ambiente di sviluppo	32
3.3	Ambiente di sviluppo: Xcode	35
3.4	Linguaggio di Programmazione: Objective C	35
3.5	Creazione del progetto	36
3.6	Pagine del Progetto	39
3.7	Lettura QR Code	42
3.8	Login	44
3.9	Registrazione	45
3.10	Scheda Locale	46
3.11	Riepilogo Ordine	48
4	CONCLUSIONI	51

5	BIBLIOGRAFIA.....	52
5.1	Libri.....	52
5.2	Siti Internet.....	52

1 INTRODUZIONE

1.1 Smartphone

La fine del precedente millennio è stata sicuramente caratterizzata da Internet: una vera e propria rivoluzione non solo tecnologica, ma soprattutto culturale. Il fatto di permettere a chiunque di pubblicare informazioni accessibili da una qualunque parte del mondo ha permesso una maggiore divulgazione delle informazioni e una migliore distribuzione e condivisione della conoscenza.

Internet ha introdotto un nuovo paradigma per lo sviluppo delle applicazioni, accompagnato da nuove tecnologie e scelte di architettura. Da programmi desktop in esecuzione sui diversi PC siamo passati ad applicazioni web accessibili da una qualsiasi parte del mondo.

Nonostante non sia passato molto tempo dall'avvento di Internet, oggi stiamo vivendo una nuova rivoluzione, quella dei dispositivi mobili: ciò che prima stava sulla nostra scrivania, ora ci può stare nel taschino. Stiamo parlando di dispositivi mobili, di quelli che prima chiamavamo in modo riduttivo con il termine di cellulare, ma che ora stanno diventando veri e propri PC portatili, dove la funzione di telefono, sebbene sia fondamentale, è solo una di quelle disponibili: questi dispositivi prendono il nome di **Smartphone**.

Gli Smartphone (in italiano telefonino intelligente) sono dispositivi intelligenti che incorporano le funzionalità di un telefono cellulare e forniscono delle funzionalità per la gestione dei dati personali.

Il numero di smartphone venduti sta rapidamente aumentando in tutti i paesi, anche grazie alla diffusione di smartphone “low cost”; con i tassi

di crescita previsti, si stima che nel giro di un paio di anni gli smartphone (che sono circa 450 milioni nel mondo) supereranno il numero di pc (pari a poco più di un miliardo); inoltre, anche il numero di utenti mondiali di mobile internet continua a crescere a tassi elevatissimi e arriverà tra pochi anni – due o tre al massimo – a superare il numero di utenti che accedono a internet da un pc.

Una delle ragioni che sta contribuendo alla crescita del mercato degli smartphone è il progressivo distanziamento tra il luogo di lavoro e l'abitazione, che ha costretto ad intensificare gli spostamenti quotidiani: oggi essi coinvolgono oltre 30 milioni di individui in Italia, di cui più di 10 milioni tra bambini e studenti e oltre 20 milioni di occupati.

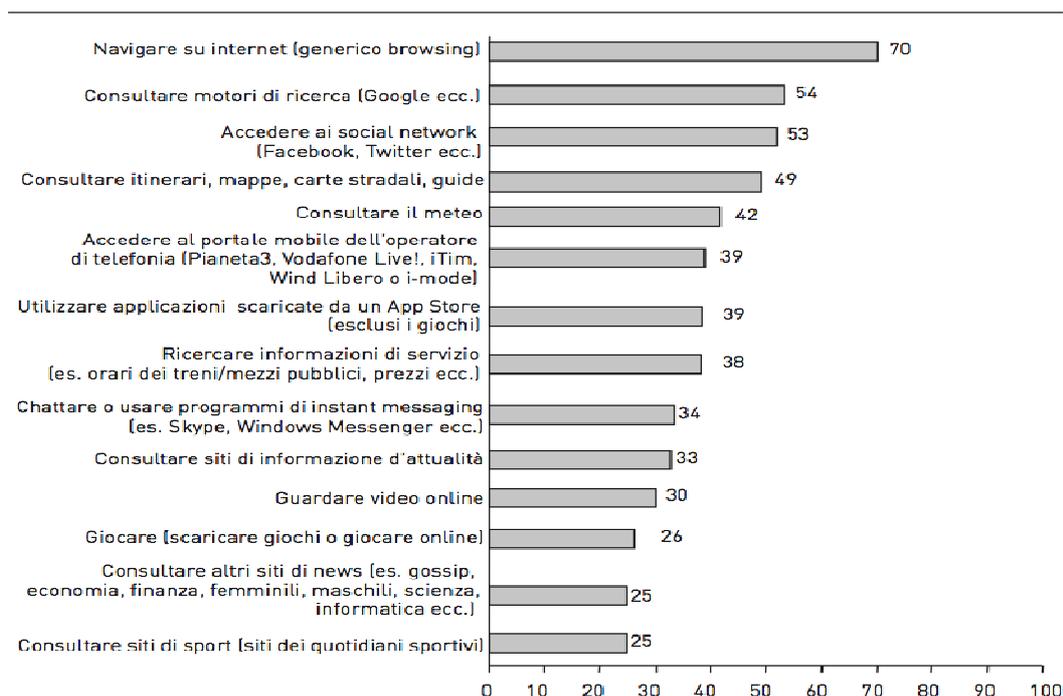
Ovviamente un PC di media potenza ha caratteristiche software e hardware (quantità di memoria, potenza di CPU e alimentazione) superiori a quelle di un qualunque smartphone attuale, anche se da tempo sono comparsi dispositivi con microprocessori quad-core, cosa impensabile fino a qualche anno fa. Si stima che la potenza di calcolo di un cellulare di nuova generazione sia paragonabile a quella di un PC di 5 o 6 anni fa; ma la distanza si riduce anno dopo anno.

I vantaggi degli smartphone rispetto ai classici pc portatili sono numerosi:

- Molto piccoli e leggeri
- Meno costosi (la maggior parte)
- Più veloci da utilizzare
- Connessione internet sempre disponibile
- Sfruttano la Localizzazione

Un ulteriore punto di forza è la possibilità di personalizzare il proprio smartphone, sia dal punto di vista fisico/estetico, ma soprattutto da quello dei contenuti e delle funzionalità. Per quanto riguarda l'aspetto delle funzionalità, infatti, viene resa disponibile una sempre più crescente personalizzazione: non solo è possibile personalizzarne toni e sfondi, ma anche la disposizione delle icone del menu. Questa tendenza, è stata resa molto più ampia dall'introduzione degli App Store (che approfondiremo nell'ultima parte dell'introduzione), che hanno in tal senso estremizzato la componente di personalizzazione del menu attraverso l'installazione di applicativi secondo i gusti e le necessità dell'utente. Per le persone, lo smartphone, costituisce uno strumento facile da usare, sempre a portata di mano ed essenziale per l'arricchimento dell'esperienza che sta vivendo: attraverso mobile internet, mobile app, realtà aumentata e confronto con terzi tramite i social network fruibili in mobilità. La figura ci dimostra come gli smartphone stiano diventando dei veri e propri sostituti dei pc portatili:

Le attività svolte su cellulari/smartphone (%)
(base: 1.043 intervistati)



Fonte: indagine Doxa per Osservatorio Mobile Marketing & Service (2010)

1.2 Sistemi Operativi

I vari smartphone differiscono tra loro per diversi aspetti:

- Sistema operativo
- Dimensioni
- Grandezza dello schermo
- Velocità del processore
- Quantità di RAM

e molto altro.

Mentre per le differenze hardware sarebbe impensabile elencarle tutte e confrontarle tra loro, ci soffermiamo su un aspetto molto importante che differenzia i vari dispositivi: il **Sistema Operativo**.

Alcuni sistemi operativi sono legati in modo biunivoco all'hardware, in quanto realizzati dal medesimo produttore: tra questi vi è Apple, che produce sia i terminali sia il relativo sistema operativo. Altri sistemi operativi supportano invece diversi terminali: l'esempio più evidente è costituito da Android, utilizzato su dispositivi di diversi produttori, quali: Htc, Sony Ericsson, Lg e Samsung; il che rappresenta un importante vantaggio in termini di potenzialità di diffusione del sistema operativo, ma anche uno svantaggio per i vari applicativi.

I principali costruttori di cellulari (Apple, Samsung, Nokia, ecc..) hanno messo a disposizione degli sviluppatori i propri sistemi operativi, ciascuno con il proprio ambiente di sviluppo, i propri tool e il proprio linguaggio di programmazione per consentire lo sviluppo e la pubblicazione di nuove App.

Ad esempio, per realizzare un'applicazione per iPhone è necessario

disporre del sistema operativo Mac OS X, e conoscere il linguaggio Objective-C.

Per cercare di comprendere meglio l'importanza di un sistema operativo sui dispositivi, vediamo nella pagina successiva una breve descrizione dei 3 sistemi operativi attualmente più diffusi sul mercato seguita da una tabella che mostra i principali vantaggi e svantaggi per l'utente nel loro utilizzo.

- **Android (Google)** : Un sistema sviluppato da Android Inc. nel 2005 e successivamente adottato da Google tramite l'acquisizione della società. L'interfaccia grafica (GUI) di Android non è unica: essa infatti è affidata al produttore del dispositivo, salvo alcune versioni che presentano la GUI nativa. Vista la notevole differenza delle interfacce utilizzate, nonché la notevole personalizzazione ammessa su di esse coi vari Widget, cartelle, icone, etc totalmente personalizzabili, si possono avere device con lo stesso sistema operativo ma totalmente diversi.
- **iOS (Apple)** : sistema sviluppato internamente da Apple solo per i propri dispositivi mobile (iPhone, iPad e iPod Touch). La prima versione risale al gennaio del 2007; dal 19 settembre 2012 è disponibile l'ultima versione, la 6.. iOS ha introdotto il multitouch, le gesture come il pinch to zoom, lo swipe per spostarsi nelle schermate e molte altre innovazioni.
- **Windows Phone (Microsoft)**: si tratta di un sistema sviluppato da Microsoft, la cui prima versione fu presentata nel 2003 e studiata in primo luogo per Pocket pc, e in un secondo momento anche per smartphone; Attualmente la versione più aggiornata è la 7.5, ma a breve è prevista l'uscita della versione 8.0.

1.3 Vantaggi e Svantaggi Sistemi Operativi

Android

- **Vantaggi:** Android può essere personalizzato: l'utente può combinare tra loro apps, cartelle e widget, e scaricare nuove interfacce dal market. La barra delle informazioni è funzionale e indica tutte le apps in esecuzione, inoltre il browser è veloce e supporta Flash. Le app possono anche interagire tra loro, inoltre l'utente mediamente esperto ha la possibilità di installare delle versioni modificate del sistema operativo in modo tale da poterlo personalizzare completamente.
- **Svantaggi:** L'utente ha bisogno di un periodo di utilizzo più lungo rispetto ad iOS per riuscire ad utilizzare al massimo le funzioni del dispositivo; inoltre deve fare attenzione in fase di acquisto al tipo di versione installata: molti smartphone montano una versione meno recente del sistema e non tutti i produttori offrono la possibilità di aggiornare all'ultima versione. Sullo Store Online si trovano applicazioni di terze parti a volte poco affidabili.

iOS

- **Vantaggi:** Molto intuitivo e facile da utilizzare: l'utente impiega poco tempo per imparare ad utilizzare il dispositivo. Le applicazioni di terze parti presenti sullo store sono molto affidabili vista la rigidità dei controlli effettuati da Apple prima della pubblicazione, inoltre l'utente può aggiornare (nella maggior parte dei casi) il proprio dispositivo all'ultima versione del sistema operativo

- **Svantaggi:** La facilità di utilizzo va a discapito della personalizzazione: sui dispositivi iOS la personalizzazione del device è pressoché nulla rispetto ad Android. Lo scambio di file, come ad esempio una canzone o un videoclip, con un altro smartphone o computer risulta molto più complicato non potendo utilizzare il dispositivo come memoria di massa.

Windows Phone

- **Vantaggi:** Maggiore semplicità e immediatezza nell'utilizzo a chi è abituato ad usare le versioni di Windows per PC, l'interfaccia è moderna e si integra molto bene coi contatti e coi social network. Possibilità di aprire, leggere e modificare tutti i documenti di Microsoft Office, suite di applicazioni molto utilizzata.
- **Svantaggi:** Il numero di app disponibili nel market è molto inferiore rispetto agli altri sistemi; le funzionalità del multitasking hanno qualche limite dovuto alla mancanza di quest'ultimo nelle versioni precedenti. Come per Android si possono trovare in commercio smartphone che non hanno installato l'ultima versione del sistema operativo e non viene data la possibilità all'utente di poter aggiornare il dispositivo alla versione più recente con la conseguenza di non poter utilizzare alcune nuove funzionalità in un terminale che sarebbe in grado di supportarle.

1.4 Mercato Sistemi Operativi

La figura sottostante mostra l'andamento del mercato dei sistemi operativi sugli smartphone Europei.

	12 w/e 04 Sep 2011 %	12 w/e 02 Sep 2012 %	Change %
GB	100.0%	100.0%	0.0
iOS	18.5	21.3	2.8
Android	49.9	62.5	12.6
RIM	22.5	10.6	-11.9
Symbian	6.3	0.9	-5.4
Windows	2.1	4.4	2.3
Bada	0.6	0.3	-0.3
Other	0.0	0.1	0.1
Germany	100.0%	100.0%	0.0
iOS	22.0	11.2	-10.8
Android	51.4	78.6	27.2
RIM	1.7	0.5	-1.2
Symbian	13.8	4.2	-9.6
Windows	7.3	4.0	-3.3
Bada	2.5	0.7	-2.2
Other	1.0	0.8	-0.2
France	100.0%	100.0%	0.0
iOS	17.2	12.2	-5.0
Android	47.8	61.5	13.7
RIM	11.6	10.4	-1.2
Symbian	10.0	2.5	-7.5
Windows	2.2	5.7	3.5
Bada	10.6	7.3	-3.3
Other	0.5	0.4	-0.1
Italy	100.0%	100.0%	0.0
iOS	18.4	14.7	-3.7
Android	37.2	58.7	21.5
RIM	4.8	3.5	-1.3
Symbian	32.6	11.0	-21.6
Windows	3.8	10.4	6.6
Bada	2.4	1.8	-0.6
Other	0.8	0.0	-0.8
Spain	100.0%	100.0%	0.0
iOS	7.3	3.7	-4.2
Android	55.2	84.0	28.8
RIM	20.2	5.5	-14.7
Symbian	14.1	3.1	-11.0
Windows	2.6	1.6	-1.0
Bada	0.0	0.0	0.0
Other	0.0	2.1	2.1

Come si nota dalla figura il sistema operativo attualmente dominante sul mercato italiano è Android.

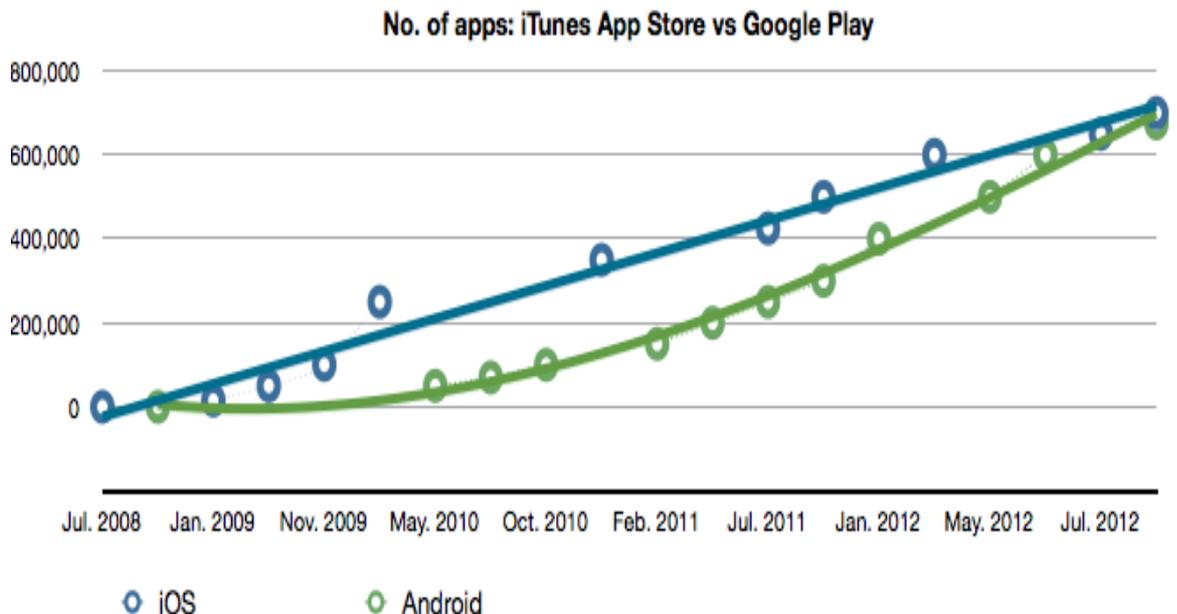
1.5 App Store

Un App Store è un negozio virtuale in cui è possibile scaricare applicazioni sviluppate da terze parti. Ogni sistema operativo rende disponibile agli utenti il proprio Store che conterrà applicazioni installabili solo sui device di quel sistema: le app disponibili possono essere a pagamento oppure gratuite e possono essere scaricate direttamente dal dispositivo. Per approfondire meglio questo argomento vediamo una breve descrizione dei due Store attualmente più importanti sul mercato:

- **App Store (Apple):** l'App Store è un servizio realizzato da Apple disponibile per iPhone, iPod touch ed iPad che permette agli utenti di scaricare ed acquistare applicazioni disponibili in iTunes Store. L'App Store è stato aperto il 10 luglio 2008 e sono disponibili oltre 700.000 applicazioni sviluppate da terze parti, con oltre 30 miliardi di download. Apple pretende di poter testare l'applicazione che si intende mettere a disposizione attraverso il suo store, per evitare di pubblicare programmi che, magari per un'errata gestione della memoria, possano fornire all'utente una cattiva interazione con il dispositivo;; se l'applicazione non rispetta alcune delle rigide regole di Apple non verrà pubblicata. Dopo il successo dell'App Store ed il lancio di servizi analoghi da parte dei concorrenti, il termine "App Store" è stato coniato per indicare qualsiasi servizio simile a quello lanciato da Apple.

- **Google Play Store (Google)** : è uno store di applicazioni, brani e tanto altro sviluppato da Google per i dispositivi Android che ha sostituito l'Android Market, il quale conteneva esclusivamente applicazioni per Android. La nascita di Android Market risale alla fine dell'agosto 2008, poco dopo il lancio dell'App Store da parte di Apple. Gli sviluppatori Android, a differenza dell'Apple Store, possono rendere le loro applicazioni immediatamente disponibili sul Play Store, senza dover attendere un lungo processo di approvazione. Attualmente il Play Store conta circa 700 mila applicazioni disponibili al download.

Nel grafico sottostante viene illustrato l'andamento del numero di applicazioni disponibili nei due store.



2 DESCRIZIONE DEL PROGETTO

2.1 Descrizione Generale

Il progetto ha come scopo quello di realizzare un'applicazione nativa che permetta ad un qualsiasi utente che utilizza uno smartphone di poter consultare il menu di un ristorante ed inviare un ordine con le porzioni selezionate.

Il progetto si divide, sostanzialmente, in due parti principali:

- Visualizzazione dei locali
- Gestione dell'ordine

L'applicazione permette la registrazione dei clienti e dei camerieri dei locali, i quali una volta effettuato il Login potranno gestire i propri ordini nel caso dei clienti, o gestire le comande attive del locale nel caso di camerieri; è possibile anche dopo la registrazione modificare i propri dati e la password dell'account. Per il momento sono attive solo le funzioni principali, ma il progetto prevede che l'app sia disponibile in multilingua per permettere l'utilizzo anche nelle altre lingue principali (inglese, spagnolo, francese) oltre che a quella italiana.

Vista l'ormai ampia possibilità di integrazione fra social network e sistemi operativi mobile, il team di sviluppo ha deciso di dare all'utente la possibilità di condividere le varie operazioni effettuate con l'applicazione; in particolare l'utente, ha la possibilità di effettuare il Login all'interno dell'app con il proprio account di Facebook e di condividere su Facebook o Twitter la propria posizione dove sta mangiando, un commento o una foto.

2.1.1 Visualizzazione dei locali

La prima parte dell'applicazione permette all'utente di selezionare un ristorante; più dettagliatamente permette di:

- Visualizzare i locali disponibili in una mappa che permette all'utente di vedere anche la posizione in cui si trova (se è disponibile la localizzazione),
- Visualizzare i locali disponibili in un elenco che, sempre se è possibile utilizzare la localizzazione, permette di mostrare oltre al locale anche la distanza di quest'ultimo dall'utente.
- Visualizzare i locali disponibili a 3 dimensioni: l'utente, ruotando la direzione del proprio dispositivo, visualizza i locali presenti in quella direzione e la distanza da essi.

Ognuna di questi diversi tipi di visualizzazione permette di filtrare i risultati in base alla distanza impostata dall'utente, che può essere cambiata in ogni momento: in questo modo verranno visualizzati solo i locali che sono nel raggio impostato. Inoltre, ognuna di queste permette di effettuare una ricerca avanzata dei locali in base ai seguenti campi:

- Ricerca per parola chiave
- Città del locale
- Fascia di prezzo media
- Tipologia del locale (ad esempio: pizzeria, ristorante, osteria, ecc..)
- Caratteristiche del locale (specialità, tipo di ambiente, tipo di cucina, altri servizi, ecc..)

Oltre alla visualizzazione dei locali, l'utente ha la possibilità di accedere ad un menu che gli permette di:

- Iniziare un nuovo ordine, selezionando il locale dalla lista dei locali che permettono il take away oppure leggendo il Qr code fornitogli dal cameriere.
- Tornare alla schermata precedente di visualizzazione dei locali (mappa, elenco o 3D).
- Aprire una pagina di aiuto
- Effettuare il Login
- Effettuare la registrazione
- Aggiornare i dati dell'applicazione
- Accedere alle Impostazioni
- Condividere su Facebook, Twitter, o via e-mail: dove sta mangiando, un commento oppure una foto da scattare o selezionata dalla libreria.

Una volta che l'utente seleziona un locale, verranno visualizzati tutti i dettagli di quell'attività, più precisamente all'utente verranno mostrate le seguenti informazioni (se disponibili):

- La foto principale
- Nome
- Indirizzo
- Numero di telefono
- Indirizzo e-mail
- Prezzo medio
- Tipi di pagamenti accettati(bancomat, carta di credito, ecc..)

- I commenti degli utenti su quel locale
- L'indice di gradimento degli utenti
- Descrizione del locale
- Gli orari di apertura
- Le news e gli eventi

Inoltre l'utente ha la possibilità di accedere al menu oppure di iniziare l'ordine in quel ristorante; il menù è suddiviso in pagine diverse a seconda della categoria della pietanza. Per ogni categoria avremo una pagina diversa, ad esempio: avremo la pagina degli antipasti, dei primi, dei secondi, delle pizze, ecc..; all'interno di ogni pagina verranno mostrate in una tabella tutte le pietanze; ad esempio, nella pagina delle pizze verranno visualizzati solamente : la margherita, la schiacciata, la capricciosa, ecc... Per ognuna delle pietanze disponibili, vengono mostrati i seguenti dettagli:

- Foto
- Prezzo
- Nome
- Possibilità di andare alla pagina di dettaglio

Nella pagina dedicata interamente alla pietanza selezionata vengono visualizzate: la foto, il nome, il prezzo, la descrizione completa del piatto, gli ingredienti utilizzati, i commenti ed i voti dei clienti su quel piatto; inoltre, potrebbero esserci anche alcuni suggerimenti ad altre pietanze che potrebbero interessare al cliente; ad esempio, se viene selezionata una pizza viene suggerita all'utente una bevanda da ordinare insieme alla pizza.

2.1.2 Gestione dell'ordine

Nell'applicazione l'utente ha la possibilità di effettuare due diversi tipi di ordine:

- **Ordine presso il ristorante** : quando l'utente arriva al ristorante e vuole ordinare, il cameriere genera dal proprio dispositivo un QR Code: l'utente dovrà fotografare il codice dal dispositivo del cameriere e, a quel punto, potrà iniziare ad ordinare dal proprio smartphone.

In base al numero di persone sedute al tavolo ed al numero di dispositivi che hanno a disposizione l'ordine si svolgerà in maniera diversa: ad esempio, se in un tavolo ci sono più persone ed un solo dispositivo l'ordine dovrà essere eseguito per tutte le persone del tavolo; nel caso in cui due o più persone abbiano la possibilità di ordinare tramite il dispositivo, ognuno può scegliere di ordinare solo per sé, oppure di ordinare anche per altri.

- **Ordine Take Away** : in questo caso l'utente può scegliere da un elenco di locali che effettuano la consegna a domicilio oppure la prenotazione e il ritiro presso il locale: dopo aver selezionato le portate e confermato l'ordine, l'utente prima di inviarlo dovrà inserire:
 - Nome e Cognome
 - Numero di Telefono
 - Indirizzo e-mail
 - Data e orario della consegna a domicilio o del ritiro
 - Eventuale indirizzo di consegna

In questo caso il locale dovrà dare conferma dell'accettazione

dell'ordine contattando l'utente via mail o telefono.

Una volta iniziato l'ordine con la scelta del ristorante o la lettura del QR Code l'utente visualizza il menu e può iniziare a selezionare le varie portate: inizialmente l'utente seleziona la categoria da visualizzare, ma ha sempre la possibilità di passare alla categoria successiva o precedente oltre che di saltare alla categoria desiderata in maniera rapida, selezionandola dalla tabella che mostra l'elenco di tutte le categorie disponibili nel menù.

Quando viene aggiunta una portata viene data la possibilità all'utente di personalizzarla, cioè di aggiungere ingredienti disponibili alla pietanza oppure di rimuovere eventuali ingredienti sgraditi; ovviamente, nel caso di aggiunte viene indicato anche il prezzo supplementare dell'ingrediente che verrà aggiunto al prezzo totale. Nel caso in cui non siano disponibili aggiunte o rimozioni, il cliente avrà comunque a disposizione uno spazio libero dove poter inserire delle note, come ad esempio il tipo di cottura o una modifica particolare che verrà letta dal cameriere.

In ogni momento il cliente potrà andare alla pagina di riepilogo dove verranno elencate tutte le portate selezionate e dove è ancora possibile effettuare modifiche quali aggiunte e rimozioni oltre che visualizzare il costo totale dell'ordine oppure decidere di annullare l'ordine e di tornare al menu principale. Inoltre, il cliente ha la possibilità di modificare l'ordine di uscita delle portate prima di confermare l'ordine, in modo che, ad esempio, una persona che ha ordinato solo un secondo non debba attendere che arrivino prima tutti gli antipasti e i primi dei componenti del tavolo, ma si può chiedere di far arrivare quel secondo insieme agli altri antipasti o primi. In

modalità ordine all'interno del ristorante, è possibile chiamare il cameriere in caso di necessità oppure inviargli un messaggio.

2.1.3 Gestione Ordini del Cameriere

Oltre alle funzionalità descritte in precedenza per i clienti, l'applicazione permette l'utilizzo anche da parte dei camerieri i quali, una volta effettuato il Login, potranno gestire tutti gli ordini attivi del ristorante; in particolare, sono disponibili le seguenti operazioni:

- **Creare una nuova comanda:** il cameriere può creare una nuova comanda inserendo: il numero del tavolo a cui fa riferimento la comanda, il numero di coperti ed il numero di dispositivi disponibili nel tavolo; se al tavolo non c'è nessun terminale disponibile il cameriere effettuerà l'ordine per i clienti dal proprio terminale. In caso ci sia almeno un dispositivo disponibile al tavolo, il cameriere genera un QR Code dal proprio dispositivo per permettere ai dispositivi dei clienti di effettuare l'ordine fotografando il codice e poi può iniziare l'ordine anche dal proprio dispositivo in caso ce ne sia necessità.
- **Visualizzare l'elenco delle comande attive:** il cameriere può visualizzare l'elenco delle comande attive nel locale e vedere lo stato in cui si trovano; per ogni comanda può: visualizzare l'elenco degli ordini di quella comanda e per ogni ordine eliminare le porzioni, visualizzare il QR Code della comanda, aggiungere un nuovo dispositivo abilitato all'ordine, aggiungere un nuovo ordine alla comanda e chiedere il conto.
- **Visualizzare le chiamate:** il cameriere può visualizzare le richieste di chiamata o i messaggi inviatogli dai vari tavoli; le

richieste arrivano al cameriere effettuando ripetutamente una chiamata al server, a cui vengono richieste l'elenco delle notifiche del cameriere: in caso ci sia una nuova notifica il dispositivo emette un suono. Inoltre il cameriere ha la possibilità di cancellare tutte le richieste ricevute.

2.2 Strutture Dati

Essendo l'applicazione nativa, i dati vengono salvati all'interno del dispositivo: la prima volta che l'app viene lanciata dopo l'installazione si avvierà il download per scaricare tutti i dati, dopodichè l'applicazione verificherà dopo un determinato periodo di tempo, se è disponibile un aggiornamento dei dati sul server e permetterà il download dei dati aggiornati all'utente; ovviamente la prima volta è indispensabile effettuare il download e quindi disporre di una connessione ad internet, mentre per quanto riguarda gli aggiornamenti, in caso l'utente non disponga di una connessione ad internet, può continuare ugualmente ad utilizzare l'applicazione.

Si è deciso di immagazzinare i dati necessari all'applicazione in file di tipo JSON (è l'acronimo di Javascript Object Notation) che è un formato adatto ad immagazzinare varie tipologie di informazioni con una struttura semplicissima che consente di capire immediatamente il significato di ciò che è scritto nel file.

La struttura di un file JSON è molto semplice: ogni oggetto può avere un nome e le diverse proprietà dell'oggetto vengono racchiuse all'interno di due parentesi graffe.

La sintassi delle proprietà di un oggetto è la seguente:

'nome proprietà' : 'valore proprietà'

ed ogni proprietà è separata dall'altra da una virgola, tranne l'ultima prima della chiusura della parentesi.

Ad esempio, se volessimo memorizzare le informazioni di una persona, la struttura potrebbe essere la seguente:

```
'persona' = {  
    'nome'      : 'Mario',  
    'cognome'   : 'Rossi',  
    'età'       : 35,  
    'sesso'     : 'M'  
}
```

Inoltre, è possibile nei file JSON nidificare più oggetti in caso avessimo più persone da memorizzare, oppure utilizzare una struttura utilizzando gli Array.

Sul dispositivo vengono salvati i seguenti file in formato json:

- l'elenco dei locali
- il menu di ogni locale nelle diverse lingue
- l'elenco delle città da poter impostare nella ricerca con le relative coordinate
- l'elenco dei filtri utilizzati per la ricerca.

2.3 Le classi del progetto

Dopo una prima analisi del problema insieme al team di sviluppo, abbiamo creato le classi del progetto, che sono le seguenti:

Categoria
nome (label visualizzata a schermo) elenco_portate (elenco ID delle portate possibili) section

Personalizzazione
nome (label visualizzata a schermo) icona tipo_scelta elenco_valori (elenco ID dei valori possibili) tipo

Valore
id_valore nome (label visualizzata a schermo) prezzo

Portata
id_portata prezzo immagine nome (label visualizzata a schermo) elenco_porzioni (elenco ID delle porzioni scelte dall'utente) elenco_personalizzazioni (elenco ID delle personalizzazioni disponibili) descrizione voto_medio calorie elenco_icone_caratteristiche personalizzazioni_default

Porzione
id_porzione prezzo elenco_personalizzazioni (elenco ID delle personalizzazioni scelte dall'utente) ordine_uscita tipo_uscita (prima di tutti o insieme agli altri)

Comanda
id_comanda prezzo elenco_porzioni (elenco ID delle porzioni scelte dagli utenti) stato tavolo numero_coperti numero_dispositivi id_cameriere

2.3.1 Categoria

Categoria

nome (label visualizzata a schermo)
elenco_portate (elenco ID delle portate possibili)
section

La classe categoria viene istanziata per ogni tipologia diversa di pietanze; ad esempio: antipasti, primi, secondi, pizza, ecc.. Gli attributi dell'oggetto sono: il nome della categoria (che verrà visualizzato dall'utente), section che è l'identificativo univoco dell'oggetto di tipo categoria e l'attributo multiplo elenco portate che conterrà l'id delle pietanze appartenenti a quella categoria.

2.3.2 Personalizzazione

Personalizzazione

nome (label visualizzata a schermo)
icona
tipo_scelta
elenco_valori (elenco ID dei valori possibili)
tipo

La classe Personalizzazione viene istanziata per ogni diverso tipo di modifica possibile alla portata; ad esempio: il tipo di cottura, la bibita a scelta nel menu, il tipo di contorno insieme alla carne, ecc.. Il tipo di scelta può essere multipla o singola; se, ad esempio abbiamo il tipo di cottura sarà una scelta singola, se abbiamo l'elenco di ingredienti da poter aggiungere alla pizza avremo una scelta multipla.

2.3.3 Valore

Valore
id_valore nome (label visualizzata a schermo) prezzo

La classe valore viene istanziata per ogni valore diverso di personalizzazione possibile; ogni valore ha un nome (ad esempio sprite, cottura media, insalata) ed un prezzo che, in caso si tratti di una rimozione sarà 0, mentre in caso di aggiunta andrà a sommarsi al costo della porzione.

2.3.4 Portata

Portata
id_portata prezzo immagine nome (label visualizzata a schermo) elenco_porzioni (elenco ID delle porzioni scelte dall'utente) elenco_personalizzazioni (elenco ID delle personalizzazioni disponibili) descrizione voto_medio calorie elenco_icone_caratteristiche personalizzazioni_default

La classe portata viene istanziata per ogni pietanza disponibile nel menu, ogni portata contiene l'elenco delle personalizzazioni disponibili per quel piatto (aggiunte e rimozioni), il voto medio dato dagli utenti al piatto, una descrizione completa ed, ovviamente, il prezzo nel menù. L'attributo personalizzazione_default indica se è possibile modificare la porzione con aggiunte o rimozioni, oppure se è presente solo il campo

per l'inserimento di testo libero da parte del cliente.

2.3.5 Porzione

Porzione
id_porzione prezzo elenco_personalizzazioni (elenco ID delle personalizzazioni scelte dall'utente) ordine_uscita tipo_uscita (prima di tutti o insieme agli altri)

Il cliente può scegliere più porzioni dello stesso piatto e non è detto che siano tutte e due uguali; ad esempio potrebbe volere 2 pizze capricciose, ma in una volerci aggiungere i funghi e nell'altra no. Per questo motivo, non basta identificare solo le “portate” selezionate dal menu, ma dobbiamo utilizzare delle porzioni, ognuna delle quali avrà il proprio prezzo base della portata più, eventualmente, le varie aggiunte scelte; quindi ogni porzione deve contenere le proprie personalizzazione ed il proprio ordine di uscita che può essere diverso; ad esempio, potremmo ordinare 2 cotolette alla milanese e volerne una insieme ai primi e l'altra insieme ai secondi.

2.3.6 Comanda

Comanda
id_comanda prezzo elenco_porzioni (elenco ID delle porzioni scelte dagli utenti) stato tavolo numero_coperti numero_dispositivi id_cameriere

Ogni utente avrà la propria comanda, mentre i camerieri potranno accedere a tutte le comande del ristorante e ne potranno modificare lo stato. Ogni comanda contiene l'elenco delle porzioni ordinate, il numero

di tavolo a cui è associata, l'identificativo del cameriere che l'ha generata ed il relativo prezzo totale.

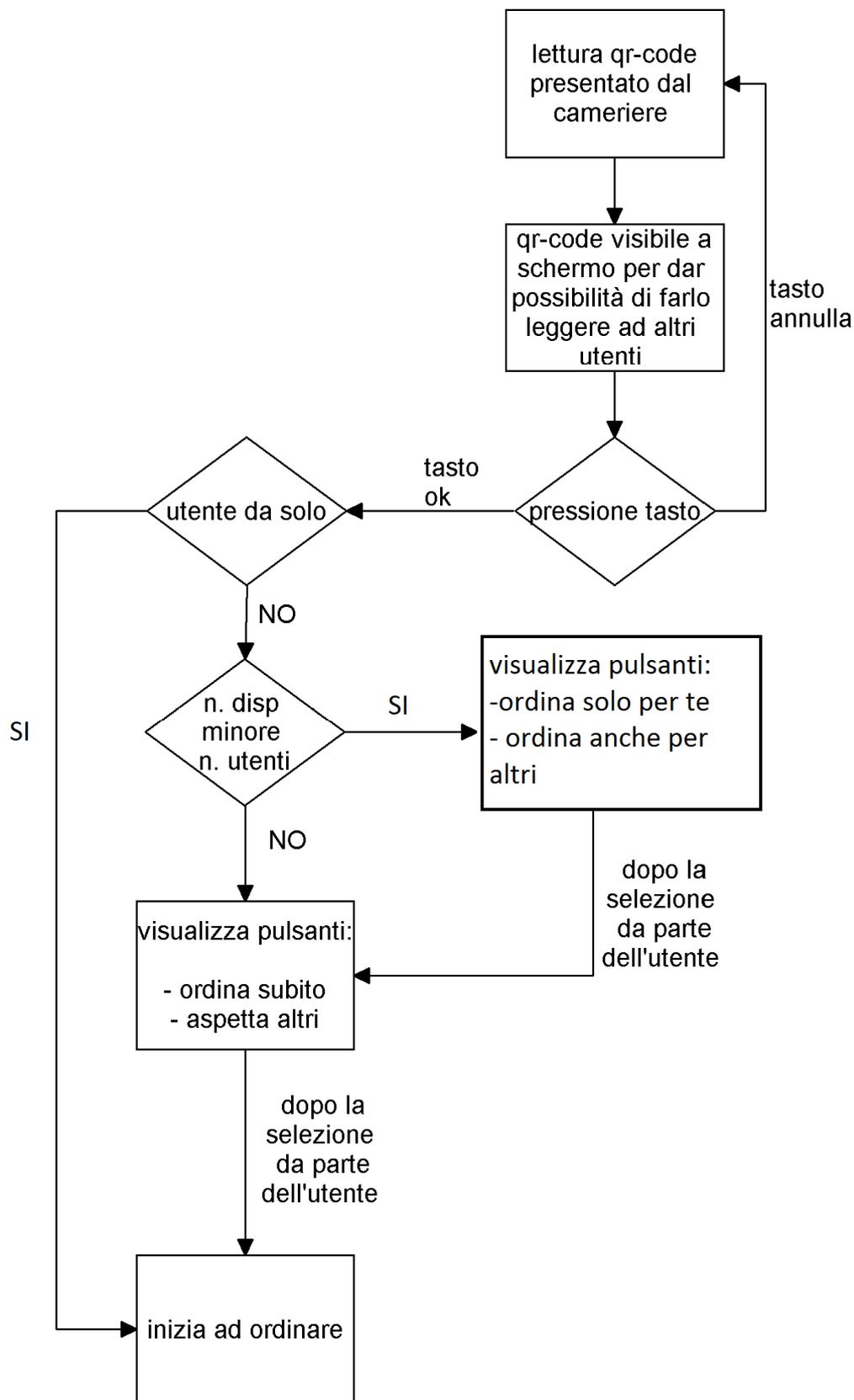
2.4 Il Server

Le chiamate effettuate dall'applicazione fanno tutte riferimento a due indirizzi che fanno riferimento ad un server : le richieste vengono inviate in formato JSON ed hanno la seguente struttura:

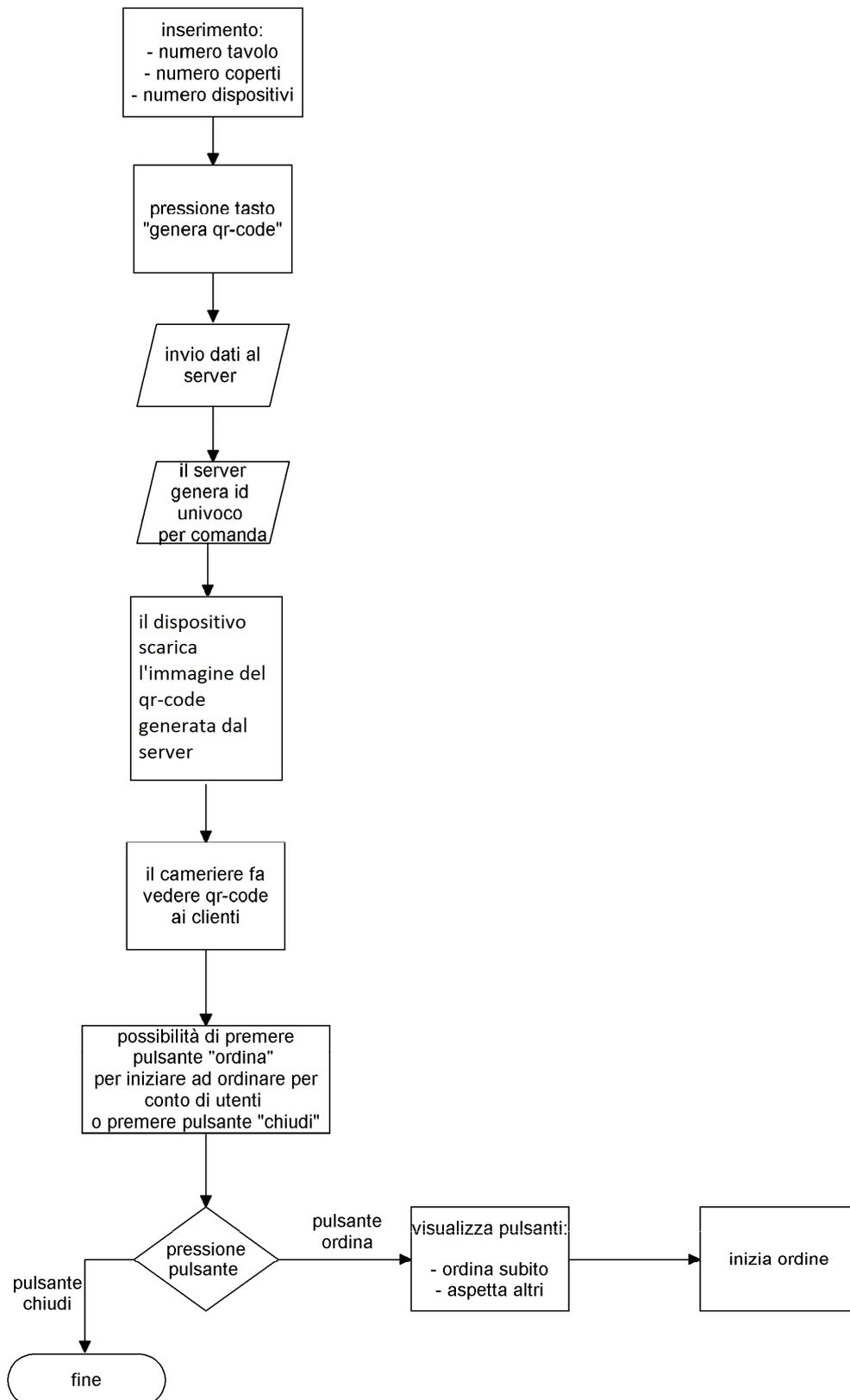
```
{
  "request": [
    {
      "action": "nome action 1",
      "data": {
        "parametro1": "valore1",
        "parametro2": "valore2",
        "parametro3": "valore3",
      }
    },
    {
      "action": "nome action 2",
      "data": [
        "parametro-elenco1",
        "parametro-elenco2",
        "parametro-elenco3",
      ]
    },
    {
      "action": "nome action 3"
    },
    {
      "action": "nome action 4",
      "data": [
        {
          "parametro1.1": "valore1.1",
          "parametro1.2": "valore1.2",
          "parametro1.3": "valore1.3",
        },
        {
          "parametro2.1": "valore2.1",
          "parametro2.2": "valore2.2",
          "parametro2.3": "valore2.3",
        }
      ]
    }
  ],
  token: "token utente rilasciato dal server se disponibile"
}
```

Le risposte del server ritornano i dati all'applicazione nello stesso formato e con la stessa logica: per ogni azione richiesta viene indicato se la richiesta ha avuto successo ed, eventualmente, gli errori verificati.

2.5 Diagramma di flusso Ordine del Cliente



2.6 Diagramma di flusso Gestione Ordine Cameriere



3 IMPLEMENTAZIONE DEL PROGETTO

3.1 Scelta dell'applicazione : Nativa o Web

Web App: E' una pagina web, accessibile da qualsiasi smartphone che supporta un web browser ed è connesso a Internet; è □ Scritta in HTML5, JavaScript e CSS. I vantaggi di una Web App sono i seguenti:

- **Costo:** Costa poco svilupparla perché non richiede grosse competenze tecniche. Inoltre vi sono dei plugin come jQuery Mobile che semplificano di molto il lavoro per coloro che non sono tanto esperti di JavaScript per la gestione degli eventi. Un evento è per esempio lo scroll, il touch su un elemento o lo swipe.
- **Veloce da sviluppare:** non richiedendo grosse competenze tecniche: sia il tempo di sviluppo che di testing è dimezzato rispetto alla creazione di un'app nativa.
- **Accessibile ovunque:** basta avere qualsiasi dispositivo connesso a internet e la webapp è possibile visualizzarla su qualsiasi device. Che sia uno smartphone, un tablet o un minipc non importa, perché le schermate si adattano automaticamente alle dimensioni dello schermo facendola sembrare una vera e propria applicazione. Inoltre l'icona sarà visualizzata nel menu delle applicazioni, proprio come una vera app.

App nativa: Le applicazioni native sono sviluppate in linguaggio macchina e devono essere scritte e compilate appositamente per ogni tipo di sistema operativo supportato dal device. □ Per esempio le applicazioni per iOS (iPhone/iPad/iPod) sono scritte in Object-C, quelle per Android sono sviluppate in Java mentre quelle per Windows Mobile

7 in ambiente .NET.

Le applicazioni native hanno numerosi vantaggi rispetto alle web app:

- **Modalità offline:** le applicazioni native devono essere scaricate e installate sul dispositivo dallo store: una volta compiuta questa semplice operazione sarà possibile utilizzarla sia online che in modalità offline: non è necessario, quindi, disporre di una connessione ad internet per poter utilizzare l'app e questo è il vantaggio più importante in assoluto; dato che non tutti gli utenti che possiedono uno smartphone hanno un abbonamento col proprio gestore per l'utilizzo della connessione ad internet, oppure potrebbero trovarsi in alcune zone non coperte dalla rete, con questo tipo di applicazione si ha la certezza che tutti gli utenti possano utilizzarla in qualsiasi momento e luogo.
- **Notifiche Push:** Le notifiche push sono dei messaggi che si possono inviare a tutti coloro che hanno installato la vostra applicazione e che ne hanno dato il consenso per la ricezione una volta installata: sono simili a degli SMS e vengono visualizzate istantaneamente sul dispositivo, risultano molti utili per inviare delle news importanti relative a qualche attività dell'applicazione, come ad esempio un offerta speciale di un'attività. Utilizzando un' app nativa è possibile inviare le notifiche all'utente che ha l'app installata.
- **Accesso a tutti i sensori** hardware installati sul dispositivo come ad esempio: accelerometro, GPS, fotocamera, magnetometro, sensore di prossimità, di luminosità, di posizionamento e tanti altri che possono essere gestiti direttamente e compiere azioni diverse in caso di movimento del dispositivo.

Il team di sviluppo ha deciso di sviluppare un'applicazione nativa per ottenere un applicativo di maggiore qualità grafica, più interattiva possibile e soprattutto che fosse accessibile anche senza la disponibilità della connessione ad internet per non avere vincoli legati alla posizione o alla rete e cercare di soddisfare qualsiasi tipo di utente che utilizza l'app.

3.2 Scelta ambiente di sviluppo

Il nostro team di sviluppo ha deciso di creare un'applicazione dedicata agli smartphone con sistema operativo iOS piuttosto che sul sistema Android per diverse ragioni:

- **Costo:** esistono moltissimi dispositivi Android, ciascuno con versioni di riferimento diverse e con caratteristiche hardware diverse; un esempio su tutto, le dimensioni e risoluzione dello schermo. Creare un'applicazione per Android che voglia abbracciare il numero maggiore di dispositivi è più costosa della equivalente per iOS.
- **Diffusione:** nelle strategie pubblicitarie, Apple punta tutto sul concetto di app, tant'è che parlando con chiunque sia al di fuori del mondo specialistico, la parola richiama subito alla mente l'iPhone e i suoi programmi. Se riesumiamo dai ricordi la pubblicità dell'iPhone 4, per 20 secondi (sui 30 della durata totale) è stato mostrato il telefono scaricare e installare un software con estrema facilità; senza contare che all'accensione del dispositivo Apple chiede di collegare una carta di credito, mentre su Android si è dovuto attendere Ice Cream Sandwich per avere un servizio del genere, e Android 4 è

presente solo sul 7% dei dispositivi, tutti di fascia alta, mentre i telefoni di fascia minore hanno quasi tutti Gingerbread (che ha una procedura un po' diversa per associare una carta di credito), e sono comunque i più diffusi, basta pensare ai vari Galaxy Ace e Next, HTC Wildfire e gli altri piccolini. iOS, invece, è disponibile su tutti i device Apple: iPhone, iPad ed iPod touch anche se le applicazioni andrebbero riadattate graficamente per ottenere un buon risultato su tutti i dispositivi.

- **Interfaccia Grafica:** Interface Builder, l'editor grafico di interfacce di Xcode, è in assoluto il migliore: l'editor messo a disposizione da Android, in confronto, risulta molto immaturo ed incompleto.

Inoltre IB risulta molto comodo anche perché l'unico layout utilizzabile, quando si sviluppa per iPhone, è quello assoluto: quindi tutti i componenti vengono posizionati ad un coordinata fissa e con larghezza ed altezza fissa.

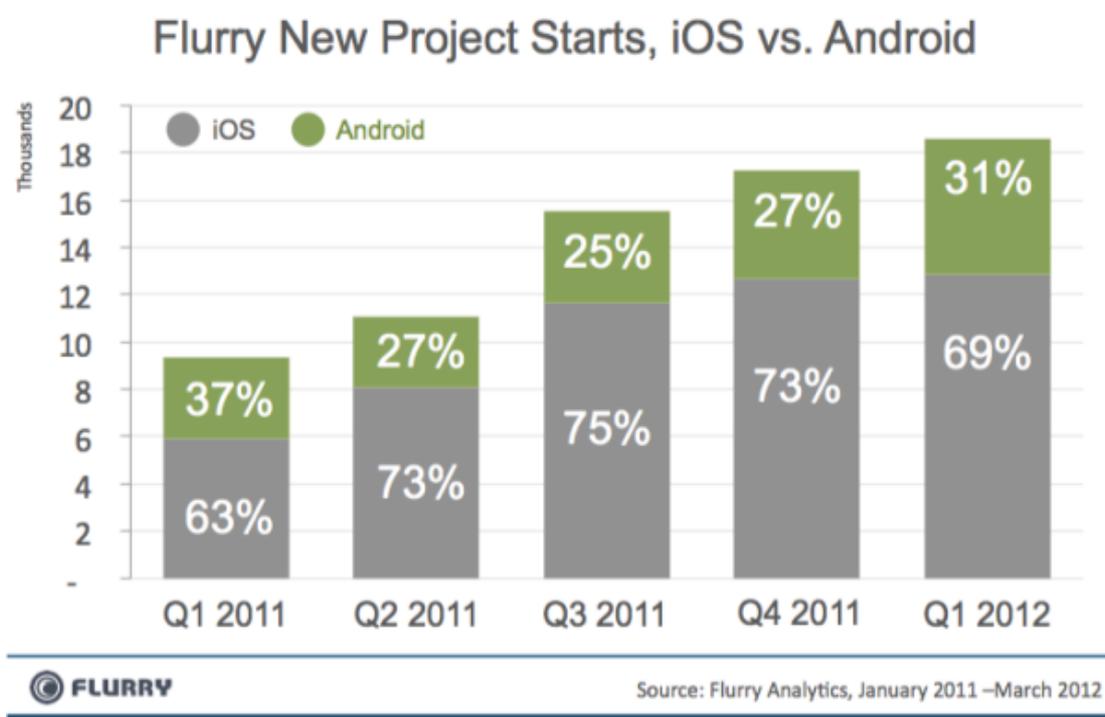
Lavorando con il tool di Android dopo qualche Drag e Drop dei componenti all'interno dell'interfaccia è necessario andare a controllare il file xml per controllare cosa sia stato generato: per questo motivo molti sviluppatori scrivono direttamente il layout in xml ed utilizzano l'editor grafico solo per avere una specie di "preview".

I componenti disponibili su iOS sono di alto livello in tutti sensi, sia perchè sono realizzati estremamente bene sia perchè sono completi e molto vicino a ciò che si aspetta l'utente finale. In pratica un progetto base "Hello world" su iOS fa già una buona impressione, magari non

fa niente, ma è già molto carino e accattivante; l'equivalente progetto "Hello world" su Android è abbastanza squallido e renderlo presentabile richiede parecchio lavoro.

Inoltre iOS mette a disposizione un simulatore molto veloce per permettere agli sviluppatori che non possiedono un dispositivo con la stessa versione software, di poter comunque testare la propria app.

Il grafico sottostante mostra la scelta fatta dagli sviluppatori di applicazioni mobile: 7 su 10 scelgono di sviluppare applicazioni per il sistema operativo iOS.



3.3 Ambiente di sviluppo: Xcode

Xcode è un ambiente di sviluppo integrato (Integrated development environment, IDE) sviluppato da Apple per agevolare lo sviluppo di software per Mac OS X e iOS. Xcode è gratuito ed è scaricabile direttamente dal sito di Apple. Estende e rimpiazza il precedente tool di sviluppo della Apple, Project Builder, che era stato ereditato dalla NeXT. Xcode lavora in congiunzione con Interface Builder (citato nel capitolo precedente), un tool grafico per realizzare interfacce grafiche. Xcode include GCC, che è in grado di compilare codice C, C++, Objective C/C++ e Java.

Una delle caratteristiche tecnologicamente più avanzate di Xcode è che supporta la distribuzione in rete del lavoro di compilazione. Usando Bonjour e XGrid è in grado di compilare un progetto su più computer riducendo i tempi; supporta la compilazione incrementale, è in grado di compilare il codice mentre viene scritto, in modo da ridurre il tempo di compilazione.

Dalla versione 3.1, Xcode è lo strumento per sviluppare le applicazioni native per iPhone e iPod touch; inoltre, dalla versione 3.2, è possibile anche sviluppare applicazioni per iPad.

3.4 Linguaggio di Programmazione: Objective C

Objective-C è un linguaggio di programmazione orientato agli oggetti basato sul linguaggio C e utilizzato principalmente per lo sviluppo di applicazioni che girano sui sistemi operativi Mac OS X e iOS. Si tratta quindi del linguaggio principe per la programmazione per Mac, iPhone e iPad. ObjC, come anche è chiamato Objective-C, non è un

linguaggio di programmazione di nuova concezione, come Java o C#, e quando si dice che è basato sul linguaggio C si intende dire che quest'ultimo è proprio parte integrante del primo. In modo simile al C++, Objective-C identifica un insieme compatto di estensioni aggiunte al C che consentono di utilizzare il linguaggio per implementare oggetti, interfacce, protocolli e altri elementi legati alla OOP. In altre parole, un programma Objective-C può contenere porzioni di codice scritti in puro linguaggio C a cui sono aggiunti classi e oggetti definiti tramite le estensioni di ObjC. Objective-C è un sovrainsieme del linguaggio C: infatti è possibile compilare qualsiasi programma C con un compilatore Objective-C, oltre che mischiare codice C con classi ObjC.

3.5 Creazione del progetto

L'unica specifica stringente per lo sviluppo di applicazioni iOS è che per poter utilizzare l'SDK ufficiale avremo necessariamente bisogno di un Mac, perché la suite di programmi per lo sviluppo gira solo su piattaforma OS X.

Disponendo di un Mac, per iniziare a sviluppare è molto semplice: basta iscriversi al programma Apple Developer (non è necessario l'account a pagamento) ed effettuare il download di Xcode. Una volta terminata l'installazione abbiamo già tutto l'occorrente per iniziare a sviluppare ed a scrivere codice.

Ogni progetto inizialmente ha solo una view principale, che è quella mostrata all'utente quando l'applicazione viene lanciata una volta termina la fase di caricamento: nel progetto si è deciso di scegliere come view principale la schermata di visualizzazione dei locali sulla mappa.

Per modellare graficamente le singole View in Xcode si possono utilizzare due diversi metodi: il primo è Interface Builder, un tool integrato all'interno di Xcode, che viene usato per la realizzazione delle interfacce grafiche, generando file .xib. L'utilizzo del tool è immediato: è sufficiente trascinare all'interno di un'area (che rappresenta una vista dell'applicazione) gli elementi grafici che si vogliono utilizzare (come bottoni, campi di testo, immagini) per poi passare al loro posizionamento. Questo strumento permette in poco tempo di ottenere buoni risultati grafici lavorando con i componenti standard messi a disposizione; l'alternativa ad interface builder è effettuare modifiche alla view direttamente da codice senza editor grafico.

Un problema delle applicazioni mobile è la gestione da parte dello sviluppatore del passaggio da una schermata all'altra: più nello specifico ci si ritrova a dover gestire il passaggio da un ViewController ad un altro in seguito ad un evento, ad esempio il tap dell'utente su un button,

Per semplificare questo lavoro ai programmatori, Apple ha introdotto un nuovo metodo per semplificare la creazione dell'interfaccia grafica e della logica di navigazione delle applicazioni iOS: questo metodo viene chiamato Storyboard.

Lo Storyboard, non consente solo di organizzare velocemente la grafica dell'applicazione ma ne definisce anche la modalità di navigazione tra i vari View Controller: con pochi clic ci troveremo una freccia che collega il controller di partenza al controller di arrivo; questo passaggio sarà collegato ad un evento.

In questo modo il programmatore evita di dover scrivere codice per ogni evento di transizione e riesce ad ottenere una visualizzazione grafica molto semplice ed intuitiva.

Il suo utilizzo ha solo un grande svantaggio: essendo un metodo introdotto recentemente da Apple, precisamente con la versione 4.2 di Xcode e la versione del sistema operativo iOS 5, si basa su molti dei nuovi componenti del SDK e quindi non è compatibile con le precedenti versioni del sistema operativo installate su alcuni device; ma, essendo molto bassa la percentuale di utenti che non ha ancora effettuato l'aggiornamento del sistema operativo (arrivato ad oggi alle versione 6.0.1) si è deciso comunque di utilizzare lo storyboard nel progetto visti gli enormi vantaggi che è in grado di fornire.

3.6 Pagine del Progetto

Le schermate visualizzabili dall'utente sono le seguenti:

- Elenco degli eventi
- Dettaglio di un evento
- Elenco delle News
- Dettaglio di una News
- Prenotazione in un locale
- Elenco dei commenti
- Scrivere un nuovo commento
- Mappa locali
- Realtà Aumentata dei locali
- Lista dei locali
- Lista dei locali Take Away
- Elenco dei filtri per la ricerca
- Elenco dei filtri avanzati
- Selezione di una città per la ricerca
- Cosa Condividere sui Social Network
- Scelta dei Social Network su cui condividere
- Login
- Registrazione
- Termini e condizioni del servizio per la registrazione
- Recupero Password per il Login
- Modifica della password
- Modifica dei propri dati personali
- Dare una valutazione al locale
- Aiuto per i comandi

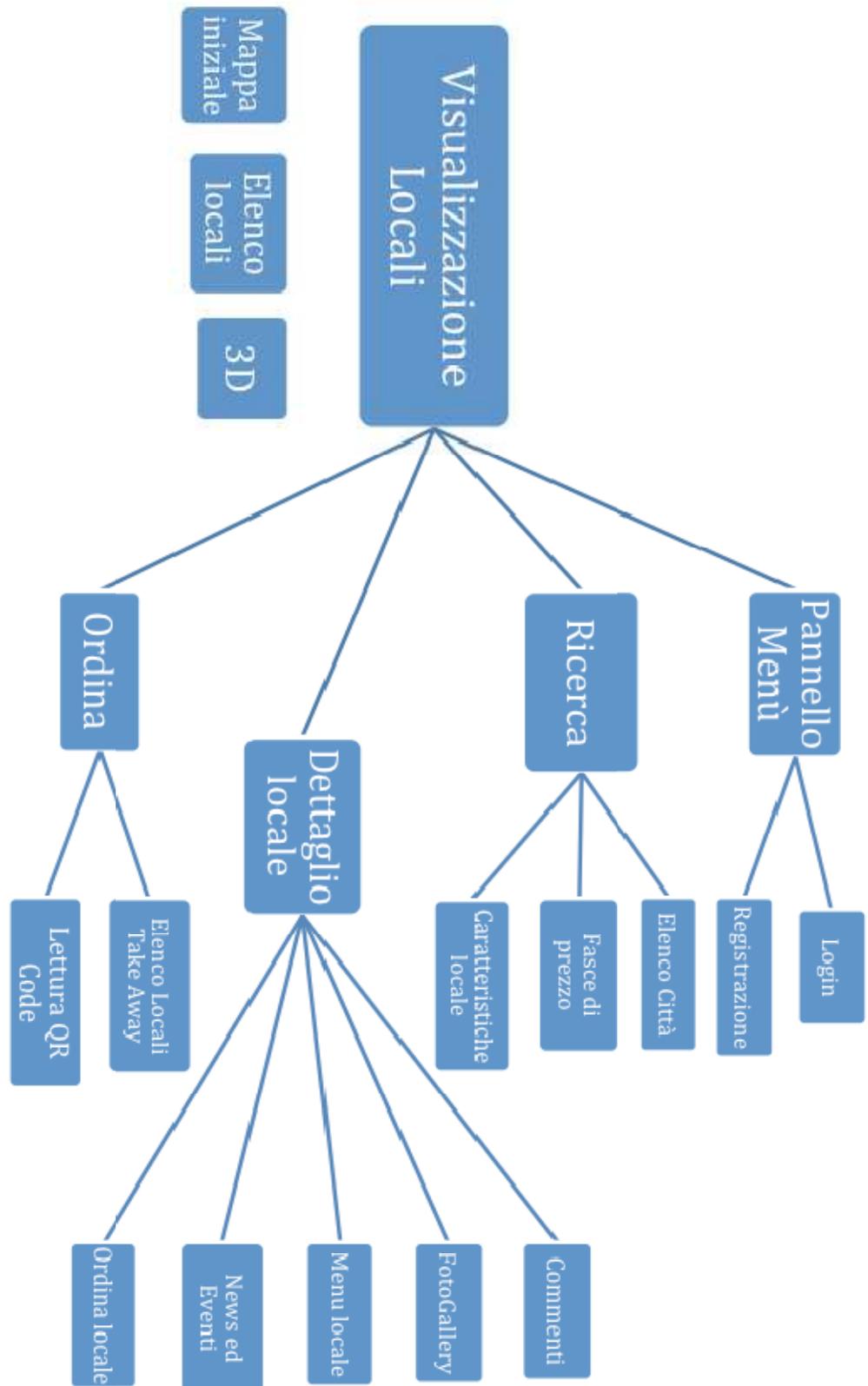
- Scheda di un Locale selezionato
- Visualizzazione Menu
- Lettura e visualizzazione QR Code
- Menu per la scelta della tipologia di pietanze
- Scheda dettagliata di una portata
- Lista delle Porzioni selezionate per una pietanza
- Modifica di una Porzione (aggiunte e rimozioni)
- Dettaglio dell'ordine
- Riepilogo Ordine
- Invio Messaggio al cameriere
- Selezione della lingua del menù
- Telecomando

Inoltre, solo per i camerieri sono disponibili anche le seguenti schermate:

- Creazione nuova comanda
- Elenco delle comande del locale
- Elenco chiamate e messaggi ricevute
- Elenco Ordini della comanda
- Elenco delle porzioni di ogni ordine con possibilità di modifica

Le funzionalità messe a disposizione da ogni schermata sono svariate, quindi approfondiremo solo le schermate più importanti e vediamo, in breve, come sono state implementate nel progetto.

Nelle prossima figura viene rappresentato lo schema di navigazione delle pagine per la parte di visualizzazione dei locali, disponibile in tre modalità: mappa, elenco e 3d; la visualizzazione di default è la modalità mappa.



3.7 Lettura QR Code

La classe `QRCodeViewController`, che eredita dalla classe `UIViewController`, si occupa di leggere un'immagine contenente un `QR Code` acquisita con la fotocamera del dispositivo; una volta letta l'immagine verrà fatta visualizzare all'utente insieme alle informazioni contenute nel codice; le informazioni contenute nel `QR Code` sono separate dal carattere underscore ('_') e contengono, in ordine:

- Codice univoco della comanda generato casualmente
- Identificativo del locale
- Codice Comanda interno del locale
- Identificativo del cameriere
- Numero del tavolo
- Numero di coperti
- Numero di dispositivi
- Tipo di Menù

Ad esempio, una stringa contenuta nel `QR Code` potrebbe essere la seguente: "1353341474_3_308_26_53_1_1_0".

Una volta verificato che le informazioni contenute nel codice sono corrette, viene creata una nuova comanda e l'utente può iniziare a sfogliare il menu e ad ordinare.

Nel caso in cui i dispositivi siano minori del numero di persone sedute al tavolo, viene chiesto all'utente se vuole ordinare anche per altri o solo per se stesso; in caso di errore di lettura del `QR Code` viene mostrato un avviso di codice non valido.

La funzione di lettura del QR Code utilizza l'SDK ZBar: integrandola nel progetto, permette di andare ad una nuova schermata con la fotocamera aperta; una volta che viene riconosciuto un codice viene fotografato e viene mandato un messaggio contenente il codice acquisito. A questo punto è possibile estrarre dal codice la stringa contenuta al suo interno: per ottenere i dati che ci interessano basta semplicemente prendere la stringa, e separare gli elementi divisi tra di loro dal carattere underscore ('_') in modo da ottenere un array che contiene tutti i campi che ci interessano nelle rispettive posizioni indicate in precedenza.

Ad esempio, per ricavarci il numero del tavolo dalla stringa, basterà prendere il quarto elemento dell'array; una volta effettuati i controlli per la correttezza della stringa, verranno inseriti tutti i campi presi dall'array nelle rispettive variabili (idLocale,idComanda, ecc..) e verrà creata una nuova comanda con i dati letti.

Dopo aver effettuato la verifica del codice ed aver creato una nuova comanda, l'utente può decidere di iniziare ad ordinare solo per se o anche per altri componenti del tavolo se il numero di dispositivi è inferiore al numero di persone; subito dopo la decisione dell'utente viene effettuata una chiamata al server per iniziare l'ordine con i parametri:

- Identificativo della comanda
- Identificativo del dispositivo
- Numero di persone nell'ordine
- Indicazione se l'utente è egoista e sceglie di ordinare solo per se

Se nel json di risposta ricevuto dal server viene indicato attraverso una variabile booleana che è possibile procedere, verrà avviato l'ordine e l'utente potrà visualizzare il menu e selezionare le varie porzioni.

In particolare vengono istanziati il view controller al centro contenente il dettaglio dell'ordine con le varie pietanze del menu e la finestra a sinistra contenente le categorie del menu che può essere visualizzata dall'utente tappando l'apposito button in alto a sinistra sempre presente nella finestra, oppure effettuando lo swipe col dito verso destra partendo dall'estremità sinistra dello schermo.

3.8 Login

La classe `LoginViewController` si occupa di far loggare l'utente all'interno dell'applicazione; sono disponibili due modalità di login:

- Utilizzare le credenziali fornite in fase di registrazione all'interno dell'applicazione : in questo caso viene inviata una richiesta al server con i parametri inseriti dall'utente; se il server accetta le credenziali, viene controllato il tipo di utente che effettua il login: può essere un utente standard oppure un cameriere.
- Utilizzare le credenziali dell' account Facebook : una volta recuperate le credenziali dell'utente viene mandato un messaggio ad un metodo della classe `FBSession` che tenta di aprire la nuova sessione.

Una volta ricevuta la risposta si ha a disposizione: la sessione, il suo stato e gli eventuali errori verificatosi durante l'apertura; quindi è possibile verificare se la sessione è stata aperta o meno.

3.9 Registrazione

La classe RegistratiViewController consente all'utente di potersi registrare gratuitamente all'applicazione; affinché la registrazione vada a buon fine dovranno essere inseriti correttamente i seguenti campi:

- Indirizzo e-mail
- Username
- Password
- Nome
- Cognome
- Anno di Nascita
- Sesso

Inoltre deve essere accettato dall'utente il trattamento dei dati personali per consentire la registrazione.

Quando verrà tappato il pulsante di registrazione verrà controllata la correttezza di tutti i campi da inserire; se l'operazione di verifica va a buon fine viene inviata una chiamata al server contenente tutti i parametri indicati in precedenza.

Se la risposta del server sarà di successo, verrà visualizzato dall'utente un messaggio che indica la corretta registrazione, altrimenti verrà visualizzato un messaggio di errore

3.10 Scheda Locale

La classe HomeSchedaLocaleViewController permette all'utente di visualizzare tutte le informazioni sul locale selezionato in precedenza.

Le informazioni vengono separate in varie sezioni per permettere all'utente una miglior visualizzazione in base al tipo di dato che si vuole mostrare; le sezioni in cui viene divisa la schermata sono le seguenti:

- Foto principale e informazioni principali del locale: nome, indirizzo, fascia di prezzo medio, tipi di pagamento accettati, voto medio degli utenti e numero di commenti; inoltre da questa sezione è possibile spostarsi alla visualizzazione del menu.
- Contatti: vengono mostrati il numero di telefono, l'eventuale numero di cellulare, l'indirizzo e-mail ed il sito internet del locale. Inoltre viene data la possibilità all'utente di: chiamare direttamente il locale, effettuare una prenotazione, inviare una mail oppure, tramite l'apertura dell'applicazione mappe, mostrare direttamente l'itinerario per il locale dalla posizione in cui l'utente si trova.
- Descrizione: contiene una descrizione completa del locale
- Orari di apertura: viene mostrata una tabella con gli orari di apertura del locale dal lunedì alla domenica, sia a pranzo che a cena
- Social: permette di visualizzare il numero di commenti e di andare alla pagina dedicata a questi, inoltre mostra il numero di foto e la possibilità di accedere alla galleria completa del locale.
- News ed Eventi: l'utente può visualizzare le ultime news o eventi, inoltre può accedere a tutto l'archivio degli eventi e delle news.

Nella parte bassa dello schermo, rimangono sempre visibili le news o eventi più recenti, che vengono fatte scorrere ogni determinato intervallo di tempo e che è possibile tappare per spostarsi alla schermata di dettaglio della news o dell'evento .



Le figure sopra, mostrano la schermata di un locale visualizzata dall'utente: come si nota dalla prima figura la foto principale viene visualizzata in modo diverso rispetto al solito: infatti questa non scompare di pari passo allo scroll in basso effettuato dall'utente, ma diviene sempre più piccola.

L'immagine viene ridimensionata in modo che sia sempre centrata e visibile fino a quando l'utente non effettua uno scrolling verso il basso molto consistente: questo è possibile controllando le coordinate del punto in cui ci si trova e, di conseguenza, cambiare la dimensione dell'immagine in base alla posizione istantanea.

Allo stesso modo, se dalla posizione iniziale ci si sposta verso l'alto, l'immagine viene allargata.

La stessa modalità di visualizzazione dell'immagine principale è stata effettuata anche sulla view che mostra il dettaglio di una portata: in particolare con la visualizzazione dell'immagine della pietanza.

3.11 Riepilogo Ordine

La classe `RiepilogoOrdineViewController` permette all'utente di ottenere un riepilogo delle porzioni selezionate.

Da questa schermata il cliente può:

- Visualizzare l'elenco delle porzioni selezionate con possibilità di eliminarle o modificarle con aggiunte o rimozioni
- Modificare l'ordine di uscita delle porzioni se sta effettuando un ordine presso il ristorante
- Visualizzare il costo totale dell'ordine ed i costi parziali delle bevande e del trasporto o del coperto, in base al tipo di ordine che sta effettuando
- Confermare l'ordine ed inviarlo

Quando il cliente tappa il pulsante di riepilogo, viene controllato se l'ordine è di tipo take away oppure se si tratta di un ordine al ristorante: in questo caso viene impostato il prezzo del coperto e moltiplicato per il numero di persone in quell'ordine : il prezzo del coperto sarà visualizzato in basso come spesa parziale già sommata al totale.

Oltre a controllare il tipo di ordine, si scorrono tutte le portate presenti nel menù e si controlla per ognuna se ci sono delle porzioni selezionate: per ogni porzione si controlla se è una bevanda e quindi il prezzo va sommato al parziale del bar, oppure se è un piatto di cucina e va sommato al totale direttamente.

Dopo aver effettuato i controlli per il prezzo totale e parziale, si gestiscono gli ordini di uscita per ogni porzione: di default ad ogni porzione viene assegnato un ordine di uscita in base al tipo di categoria alla quale appartiene; l'utente ha, comunque, la possibilità di modificare l'ordine di uscita delle varie porzioni, può anche eliminarle definitivamente dall'ordine oppure modificare le eventuali variazioni effettuate sulla porzione.

Quando l'utente decide di confermare l'ordine e di inviarlo, gli viene chiesto se vuole attendere gli altri in caso ci siano ulteriori dispositivi che stanno ancora ordinando; prima di inviare l'ordine viene chiesta una conferma all'utente per l'invio.

A questo punto, a seconda del tipo di ordine avremo due situazioni diverse:

- Se l'ordine è di tipo Take Away: il cliente deve inserire i propri dati personali (nome e cognome, telefono, indirizzo e-mail), la data e l'ora della consegna o del ritiro presso il locale e l'eventuale indirizzo di consegna.

Il cliente può contattare il locale telefonicamente, oppure decidere di confermare i dati inseriti ed inviare l'ordine: in questo caso l'ordine verrà inviato al server ed il suo stato sarà "in attesa di conferma" da parte del locale, che contatterà il cliente dopo aver accettato l'ordine e cambiato il suo stato in "confermato".

- Se l'ordine è all'interno del locale viene inviato direttamente al server e non necessita di conferme; i camerieri del locale potranno visualizzare l'ordine e modificarlo.

4 CONCLUSIONI

La tesi aveva come obiettivo quello di sviluppare un'applicazione iOS che non esistesse ancora sul mercato, cosa molto complicata dato che al giorno d'oggi, esistono applicazioni praticamente per ogni tipo di cosa, utile o inutile che sia.

Il progetto è nato da un'idea innovativa e molto più generica, inizialmente poco più di una bozza, che col passare del tempo è stata migliorata e più mirata ad un obiettivo più specifico.

L'applicazione si può ritenere già abbastanza completa viste le tante funzionalità che mette a disposizione del cliente e dei camerieri dei locali, ma questo non significa che non ci siano possibilità di migliorarla: ad esempio, l'utilizzo nelle lingue diverse da quella italiana è ancora poco sviluppato ed è un aspetto molto importante, che permetterebbe una diffusione molto più veloce dell'applicazione e la possibilità di utilizzo da parte di qualsiasi tipo di utente.

Considerando la scarsa esperienza in questo campo, le difficoltà incontrate non sono state poche, ma questo mi ha permesso di imparare molte cose nuove sia a livello umano, trovandomi per la prima volta a sviluppare con altre persone, sia a livello pratico per lo sviluppo dell'applicazione e la scelta di alcune modalità piuttosto che altre.

Questa esperienza risulterà sicuramente utile in una futura esperienza lavorativa ed anche a livello personale, perché il portare a termine questo progetto dimostra che con l'impegno e la forza di volontà si possono ottenere buoni risultati e grandi soddisfazioni.

5 BIBLIOGRAFIA

5.1 Libri

1. ,D.Mark, J.Nutting, J.LaMarche, Beginning iOS 5 Development , Apress, 2010
2. J.Steele, N.To , The Android Developer's CookBook, Addison-Wesley, 2010
3. S.Grimes, C.Francis, iOS 5 Recipes, Apress, 2012
4. M. Banzi, Android Guida per lo Sviluppatore, Apogeo, 2010
5. A. Allan, Learning Apple iOS Programming From Xcode to App Store, O'Reilly, 2011
6. Punto Informatico, iPhone La tua Prima app, Edizioni Master
7. Prof. Mirko Ravaioli, Dispense del corso di Mobile Web Design, 2011/2012
8. A.Boaretto, G.Noci, F.M.Pini, Mobile Marketing, Gruppo 24 Ore, 2011

5.2 Siti Internet

1. 10 ragioni per scegliere Windows Phone,
<http://www.creagratis.com/ragioni-scegliere-windows-phone-7>
2. meglio programmare in ios, windows phone o android,
<http://blog.setale.me/2012/04/03/meglio-sviluppare-per-ios-android-o-windows-phone/>

3. sviluppare su android conviene?,
<http://www.androidworld.it/2012/06/15/editoriale-sviluppare-su-android-conviene-92870/>
4. da java e android a objective c e iOs andata e ritorno,
<http://www.cosenonjaviste.it/da-java-e-android-a-objective-c-e-ios-andata-e-ritorno-parte-i/>
5. grafico programmatori ios vs android,
http://static.blog.it/melablog/flurry_ios_android_sviluppatori_2012.png
6. app nativa o web app? ,
<http://blog.tagliaerbe.com/2011/05/app-nativa-web-app.html>
7. smartphone ,
<http://it.wikipedia.org/wiki/Smartphone>
8. Objective-C, Xcode, Interface Builder e Simulatore.
<http://www.html.it/pag/32831/objective-c-xcode-interface-builder-e-simulatore/>
9. Hello iPhone,
http://education.mondadori.it/allegato/?tipo=DCE&codSito=MI&codSoc=55&categObj=DOC&cod_allegato=5961&nvpic=dce_5961_capitolo_di_esempio.pdf
10. Lo Storyboard,
<http://www.mobilewd.it/hcDownloadAllegato/id/.../tesina.html>
11. Introduzione all'uso di Storyboard,
<http://www.devapp.it/wordpress/t103-introduzione-all-uso-di-storyboard.html>