

Success Story

“Enosis sets new standards in the mobile application domain by designing an iPhone & iPad application that gives consumers the freedom to choose restaurants that match their preferences and proximity.”



Background

To address the needs of the food service industry the client has been developing a customized Business Intelligence Web based tool in collaboration with Enosis that will use the client's high end predictive modeling technology **to connect the right customer to the right restaurant with the right discounts**. The client wanted us to design an application for iPhone and iPad devices with the same functionalities so that any consumer could reap the benefits of the services at any time and from any location. Similar to the Web Application, the mobile application would match individuals to the best local restaurants via offers, events and promos. There was the need of a system which would be strongly focused on performance and yet be very easy to use.



The Client

SpoonByte is a US based “software on demand” service provider offering iPhone & Web app that allows restaurants to create promotions (e.g., meal discounts, menu specials, or events) to increase revenue. It’s the first app that serves the food industry by focusing on *Match-Based Deal Dissemination* for the physical marketplace.

SpoonByte intelligently sends each promotion only to consumers that are near the restaurant and are predicted to act on the promotion and visit the restaurant.

CHALLENGES

- Defining an architecture to satisfy the application’s ambitious requirements. The application must function as both a native application as well as a web application working from the iPhone
- Switching from .NET framework to Cocoa
- Using Objective-C to develop the application was a challenge since it is different from the traditional languages in use such as C#, Java etc.
- Objective-C differs from traditional languages in
 - Memory management.
 - Concept of message passing.
 - Language syntax.
- Designing the solution to achieve extremely low latency in information retrieval and high throughput
- Sending notification to the application from server
- Creating a page/data refresh mechanism in device so that users can view and select updated deals and offers
- Support data persistence when server access is unavailable (Server is offline)
- Show geographic information about restaurants on a map
- Present information, customized by user's current location

ENOSIS’ APPROACH

The client collaborated with Enosis to build an application to address these extremely challenging requirements. Enosis designed a powerful and flexible application that can be useful to consumers in their daily life for connecting to the restaurants that match their choice of food and budget and at the same time help restaurant owners fill up empty tables. The solutions to each challenge were provided by a well-planned and proactive approach which involved an efficient application development model.

There are three types of iPhone applications:

- Basic iPhone application, which is an application developed using only the iPhone SDK to run natively on iPhone OS-based devices.
- Web-only content, including web applications, which are websites that are customized to behave like built-in iPhone applications.
- A hybrid application, which is an iPhone application that provides access to web content primarily through a web-content viewing area (UIWebView), and includes some iPhone OS user interface elements.



SpoonByte in iPhone

After thorough analysis of the requirements Enosis chose to build the application as a hybrid having

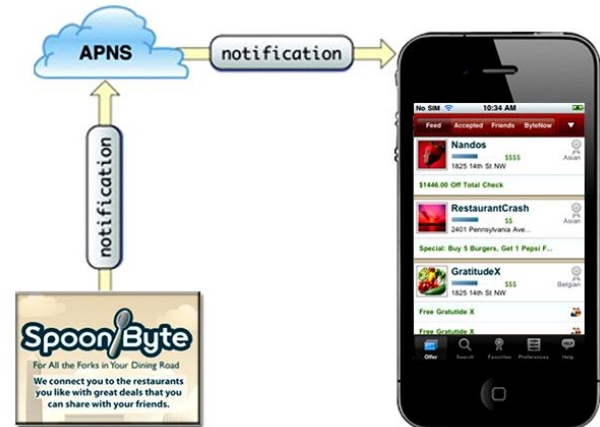
both native iPhone interface elements and web application features. This was done because of two major reasons:

- Contents can be shown in WebView using html. Enosis’s team of developers has extensive experience and knowledge of Html, CSS and Javascript.
- UIWebView supports additional CSS rules (WebKit rules) that help to showing high end graphics with little effort.

Even though the application was hybrid, Enosis wanted to make it look native so that the application became more user-friendly for regular iPhone users. That is why all of the navigation controls were native. Enosis used UINavigationController and UITabBarController for navigation.

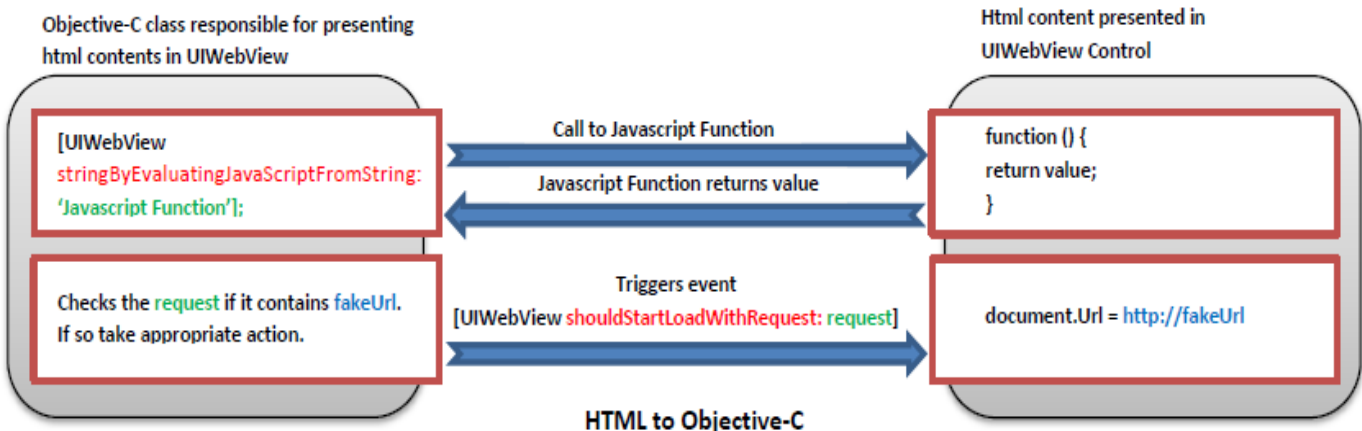
Loading html pages from the server can result in slowing the application down significantly especially when many users log in simultaneously. Enosis solutions designed a methodology to overcome this problem. All of the Html pages were in a local file system. When loading a page in UIWebView, AJAX request is sent to server, which returns JSON data and that data is loaded to the page using Javascript.

control. Information about changes was sent by Javascript by raising an event to UIWebView delegate.



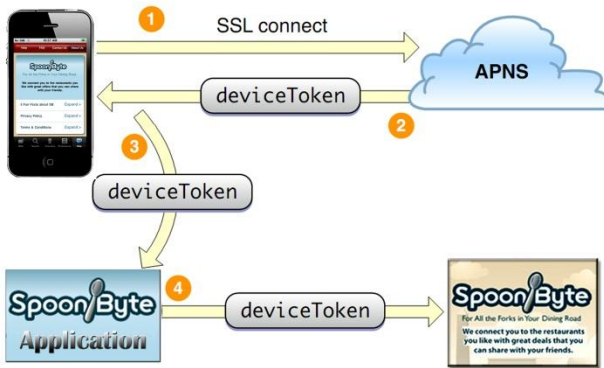
Apple Push Notification

Enosis solutions used Apple Push Notification Service to send notification and alerts. It is a robust and highly efficient service for propagating information to devices such as iPhone, iPad, and iPod touch devices. Each device establishes an accredited and encrypted connection with the service and receives notifications over this persistent connection. If a notification for an



Next a process needed to be implemented to efficiently communicate between Cocoa and Javascript. Information was pulled from UIWebViewControl by executing a Javascript function which sent a message to UIWebView

application arrives when that application is not running, an application badge number is updated and an alert appears.



Apple push notification process

To refresh page and data, Enosis used well-known “pull down to refresh” control. This is used in common applications like Facebook. Moreover the option of refreshing page from navigation stack or from tab bar controller whenever a view is presented was also included.



Pull down to refresh feature

To support data persistence when server access is unavailable Enosis used LocalStorage (A key-value pair dictionary data structure) which a feature of HTML 5.

By integrating Google’s Map Api the visual geographic information was presented and to present information customized by user's current location iOS's 'Core Location Framework' was used.

The Core Location framework can determine the current location or heading associated with a device.



iPhone showing map

DISCERNIBLE BENEFITS:

- The application achieved and exceeded every performance target specified in the requirements.
- It was user-friendly and accurately determined user preferences based on location.
- This application allowed consumers to receive the core features of the Web Application while being mobile.
- Since information from the application mostly uses the storage available in the main server it does not take up space in the device.

TOOLS AND TECHNOLOGIES

- Programming Language:** Objective C, Javascript
- Framework:** Cocoa
- Operating system:** MAC OS
- Platform:** Xcode
- Data transfer protocol:** JSON
- Web Programming languages:** AJAX, HTML, CSS