



Global KTech - Enterprise Mobile App – Case Studies

Purpose

The purpose of this document is to showcase the relevant experience.

FenceOutlet Case Study

Introduction

Fence Outlet is a fencing expert company in Florida. We built distributed iPad application with the supporting technology to allow the fence company staff to make proposals and get payments right on the field with customers.

The Problem

This system replaces the existing paper work which was not only cumbersome but also error prone due to a large computation to be done with complex pricing algorithms.

Working towards a solution

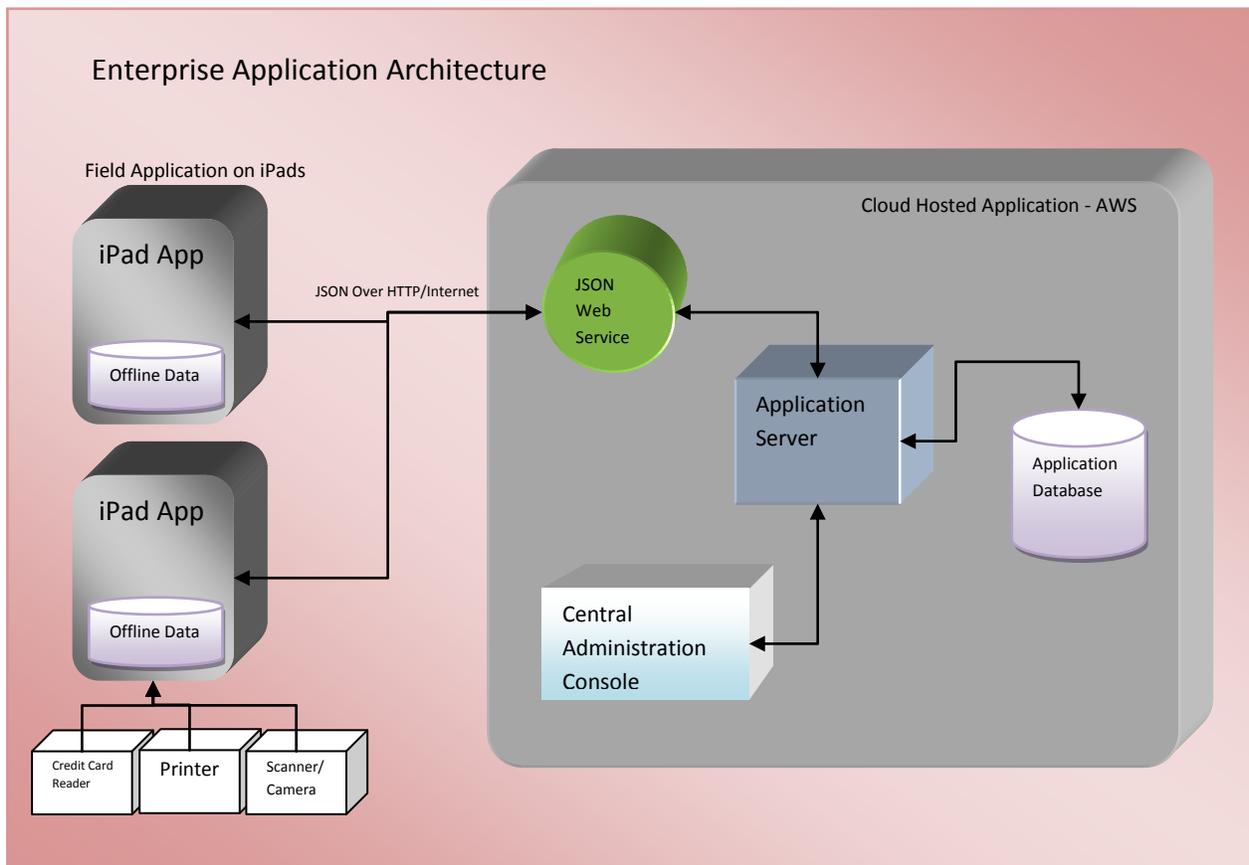
GlobalKTech's approach to the challenge ahead lay in a phased architectural re-factoring approach. The system was built from scratch, in a way that ensured maximum productivity and scalability.

- ✓ The first step was to build a shared understanding of the problem and required solution. Interactive exercises, discussions and quick prototype development fed into the project roadmap, identifying areas of technical risk and developing mitigation plans.
- ✓ A well-defined roadmap clearly identifying risks and challenges ahead of their occurrence enabled collaborative decision-making and prioritization by key stakeholders.
- ✓ Dedicated video-conferencing, use of GotoMeeting, Assembla Collaboration tool and Instant Messenger with regular meetings by GlobalKtech and fenceoutlet staff between US and Pune ensured open and frequent communication. Communication is the key to successful projects
- ✓ Production environments were simulated to enable real-time integration and testing of software under development.
- ✓ Software build and deployment processes were automated so that the team could manage several versions of the software efficiently.

- ✓ An automated regression test framework created a safeguard around the software, rapidly identifying faults and ensuring the highest levels of quality across the application. This also allowed the application to scale rapidly and handle heavy memory conditions.
- ✓ Software development was completed in small iterations, tackling the most important, high risk or troublesome areas first. Every two weeks, functionality was “showcased” to the customer to review and re-prioritize work for the next iteration. Agile methodology was followed to ensure quality and timely delivery.

The Solution

We proposed and developed an enterprise grade iPad application from scratch to allow the fence proposal makers to do all tasks with a single tool. It allows the users to use the proposal making and estimating application to work even in offline scenarios which are very common in this business.

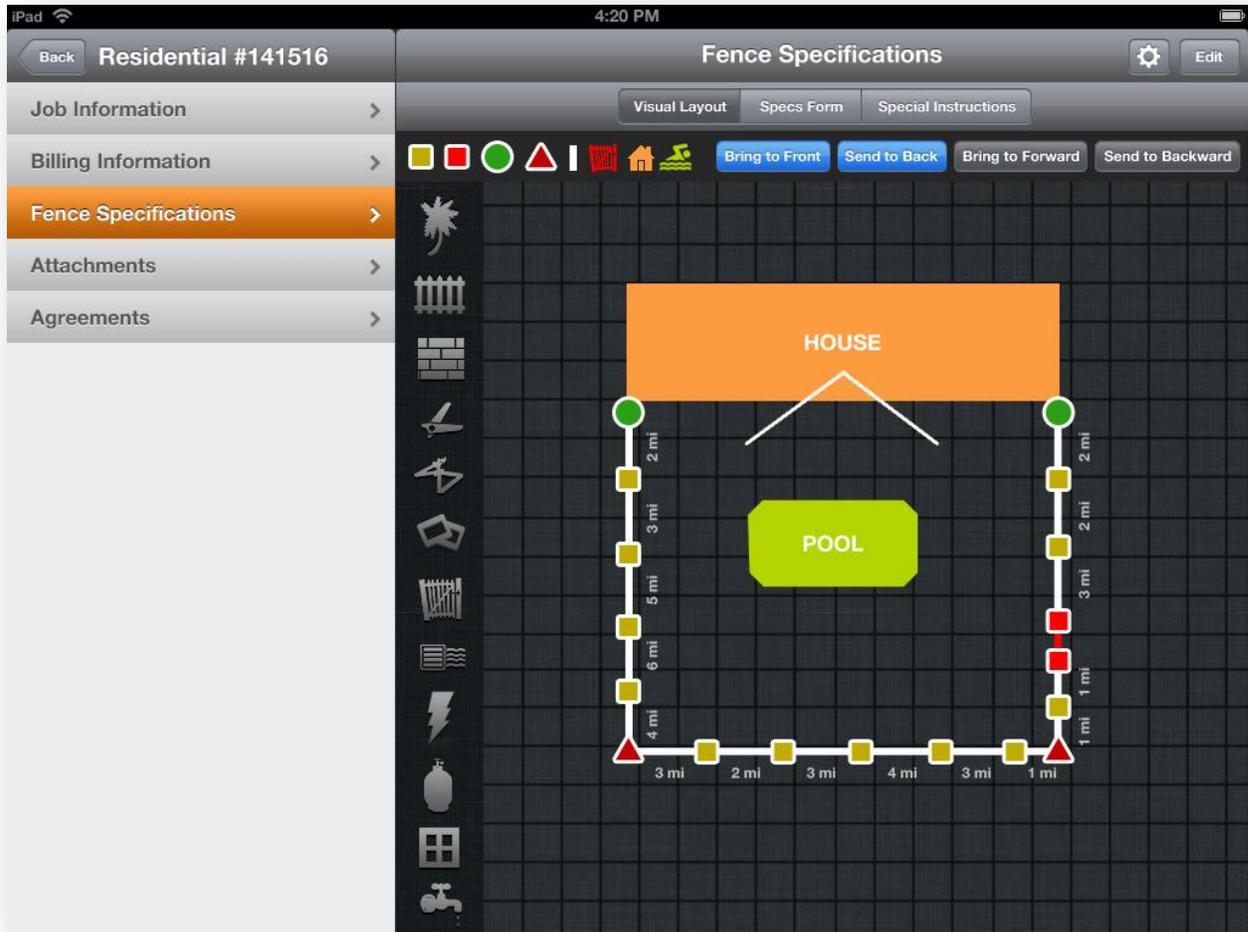


Sample Screenshot of iPad:



The Design Surface and the Price Estimation Automation

The designing of the fence involves several artifacts like the material of the fence, the type of fence, other fence options and gates with multiple options. The price depends on the selected options, labor costs and sales tax.



Cost Calculation Formula (Pseudo Code/Algorithm)

The total price of any component on the spec form is calculated as follows :

$((\text{material cost} + \text{sales tax}) + \text{labour cost}) / \text{markup}$.

The sales tax is fixed based on the store location. Orlando is 6.5%, Oviedo is 7% and Tampa is 7%. All pricing data is stored in local database and synched with server when a new proposal is downloaded.

Formulas:

1. Style Costs

$$\text{SDC} = \text{markup} + \text{labor cost}$$

(SDC - Style Dimension Cost)

2. Component Cost

```
for each component
{
  CC= CC + quantity * price;
}
```

3. Options Cost

```
for each option
{
  if(Option Selected)
  {
    OC=OC + option labor cost + material cost
  }else{
    OC=OC + Default options price
  }
}
```

4. Total Cost

Final Cost = SDC + CC + OC

Sencha Touch Experience

Technologies used in P4H Project

1. Groovy (with grails plugins for server side scripting).
2. Sencha Touch 2 (HTML 5 Library used for mobile cross platform application).
3. MySQL (Database).
4. JSP and Servlet (Website).
5. Database migration plugins.
6. PayPal plugins.
7. GIT (Version control).

Sencha Touch Experience:

1. User interface
 - a. MVC Architecture
 1. Models
 2. Controllers
 3. Views (all UI controls)
 - b. AJAX to call remote web services
2. Database
 - a. Stores (for local storage of relational data)
3. External Plugins
 - a. Refresh plugin
 - b. Pagination plugin
4. Native Packaging using phonegap