Delivering Software to Customers
Srinath Vaddepally
CEO & Founder
Srinath.Vaddepally@ristcall.com
Cell : (816) 728 2134

www.ristcall.com
Agenda

• Expectations & setting up the stage
• Quick Intro to RistCall
• Product development phases
• Software delivery models
• IoT enabled SaaS model design considerations
  • Client device s/w management
  • Server s/w management
  • Hardware device management (HaaS)
Expectations

• Overview of Software delivery models
• Design architecture considerations
Patient nurse communication issues

Bedside technologies

Nursing Workflow tools

Administrative tools
Poor Safety, Experience and Productivity outcomes

1. Safety: Falls, Infection spread, Pressure ulcers
2. Experience: HCAHPS, QAPI scores
3. Productivity: Hospital re admissions, length of stay
Ristcall wearable communication platform

Administrative Dashboard

Patient device

Staff device: RN, CNA, LPN
Project Status

Outcomes achieved

Testimonials received

Resident feedback

Staff feedback

Charles Morris
Nursing home

30

12
System modules

1) Client devices
   users : patients, nursing staff

2) Admin Dashboard
   users : CNO’s, DoN’s
DIVIDING LINE
EARLY-STAGE STARTUPS LOCAL SUPPORT RESOURCES

CULTURE • OPPORTUNITY • TALENT

INCUBATORS • FUNDING LATER-STAGE RESOURCES

COMMERCIALIZATION

IDEA R&D TECHNOLOGY TRANSFER PRODUCT LAUNCH PRODUCT SUCCESS BUSINESS SUCCESS

MENTORSHIP
FOUNDER'S TOOLKIT
RESOURCES

VALEY OF DEATH
Software Delivery Models

1) On Premise Software Deployment

“Software that is installed and operated on computers located on premise of software licensee, rather than remote facility”

Ex: ERP (Enterprise resource planning), CRM (Customer resource management)

Limitations: Expensive, Limited customization etc

Software Delivery Models

2) Hosted Software Deployment

“Software that is physically resides at a third party data center – hosting site – where it is operationally managed by service provider on behalf of license”

Ex: ASP (Application service providers)

Limitations: Single tenant, Significant over head cost

Software Delivery Models

3) As a service (SaaS)

“Software that is shared and scaleable that customer access over internet using web browser or mobile device”

Ex: Web enabled applications

Limitations: Security, IP etc
## Software delivery models

<table>
<thead>
<tr>
<th></th>
<th>On Premise</th>
<th>Hosted Software</th>
<th>As a Service model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development</td>
<td>One size fits all</td>
<td>Modifies on premise for online delivery</td>
<td>Ground up development for online delivery</td>
</tr>
<tr>
<td>Deployment</td>
<td>Installed on customer H/W</td>
<td>Installed on vendors hardware</td>
<td>Installed on vendors hardware</td>
</tr>
<tr>
<td>Implementation</td>
<td>Lengthy</td>
<td>Lengthy</td>
<td>Quicker implementation</td>
</tr>
<tr>
<td>Integration</td>
<td>Difficult and expensive</td>
<td>Difficult and expensive</td>
<td>Via API’s (Application programming Interfaces)</td>
</tr>
<tr>
<td>Multi tenancy</td>
<td>NA</td>
<td>Single tenant</td>
<td>Multi tenant</td>
</tr>
<tr>
<td>Hardware requirements</td>
<td>Operating system dependent</td>
<td>Can be Operating system dependent</td>
<td>Delivered via Web browser</td>
</tr>
<tr>
<td>Up grades</td>
<td>Once a year</td>
<td>Dependent on development cycle</td>
<td>Monthly or frequently</td>
</tr>
</tbody>
</table>

Architecture design considerations for “IoT enabled SaaS model”
Ristcall wearable communication platform

Administrative Dashboard

Patient device

Staff device: RN, CNA, LPN
Software components

- **Client device software modules**
  - Android OS
    - Full blown OS
    - Battery optimized OS
    - Semi optimized OS
  - Android Application

- **Server & Web dash board modules**
  - Back end server logic
    - Cloud server : Amazon Server
    - Database : MySQL, SQLite
  - Front end web dash board
    - Application server : Tomcat
    - UI Technologies : Spring frame work, JDBC, Java, J2EE, Web sockets, Maven
IoT based SaaS product management

- Client software management
- Server software management
- H/W Device management using software (for HaaS – Hardware as a Service)
Client Software Management
Client architecture design considerations

• Hardware might change
  • Android, iOS, Windows, Tizen based wearables

• OS might change
  • Full blown OS – Play store availability
  • Battery optimized OS – No Playstore / App store
  • Semi optimized OS
    • Device root certification
    • No Device root certification

Create “hardware abstraction layer” to make Hardware agnostic
Client architecture design considerations

• Software product “updates”
  • Playstore update
  • Remote update

• Software product “upgrades”
  • Fall detection, wander detection

• Battery optimized software
  • WiFi On /Off
  • Screen time out
  • Sleep activity

• Touch optimization
  • Single touch
  • Single tap
  • Constant press

• Wireless Communication
  • WiFi / Bluetooth antenna
  • Socket Vs TCP Vs UDP communication
Cloud server stability tests

- **Load Testing**: For a given hardware configuration, ensure that the server is responding to all the requests from patients and nurses devices.

- **Capacity Planning**: For the given load conditions (both present and future), determine the optimal hardware configuration.

- **Business Continuity**: What happens when a single server goes down? (Single tenant vs multi tenant)
Cloud server performance metrics

• Call turn around time : 15s

• Server miss rate : 1 in 1MM

• Battery call turn around time: 40s
Device management issue

• Protection against theft in facility
• Clean ability of devices
• Device tracking
• Battery management
• Product usage / metrics
Questions ??