

CaptainCasa Enterprise Client RISC

...in a Nutshell...

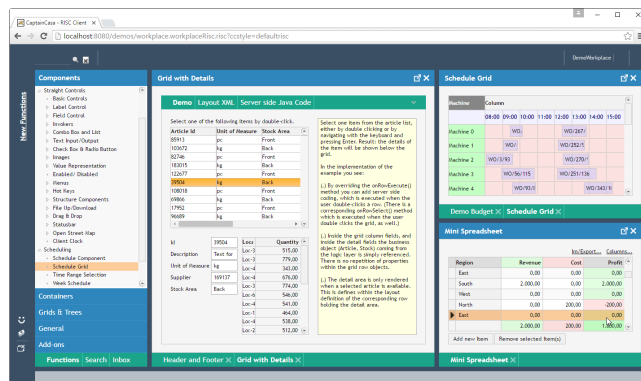
POSITIONIERUNG

CaptainCasa Enterprise Client ist ein HTML Rich Client Framework für anspruchsvolle, operativ genutzte Geschäfts- und Unternehmensanwendungen.

Merkmale solcher Anwendungen sind:

- hohe Anzahl an Dialogen
- komplexe, Server-seitige Logik
- hohe Anforderungen an die Ergonomie und an die Performance, die den Ansprüchen professioneller Benutzer (z.B. Sachbearbeiter) gerecht werden
- lange Lebenszyklen, verbunden mit hohen Effizienzanforderungen an die Entwicklung und Wartung

CaptainCasa Enterprise Client basiert im Frontend auf einem HTML(5) Client. Die Anbindung der Anwendung geschieht im Backend über J2EE Standards.



Demo Workplace

Durch Verwendung der sog. HTML-RISC-Methode werden Kernprobleme der Oberflächenentwicklung im Browser per Architektur gelöst:

- Exzellente, langfristige Browser- und Device-Kompatibilität
- Hervorragende Performance, auch bei komplexen Szenarien
- Einfache, effiziente UI-Entwicklung und Anwendungsanbindung.



Long term
Browser / Device
Compatibility



Overcoming
Browser Layouting
Limits



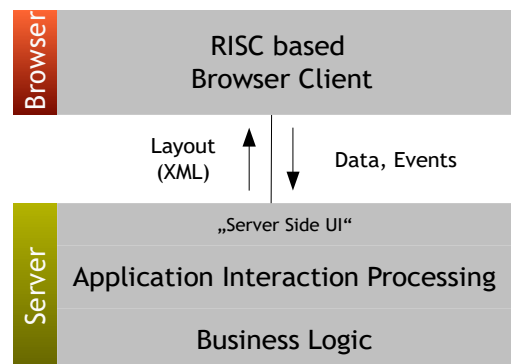
Industry-
Performance
and
Robustness

BUILDING BLOCKS

CaptainCasa Enterprise Client folgt dem Prinzip der Server-seitigen Oberflächenentwicklung: Dialoge werden vom Server als XML Beschreibung zum Browser gesendet. Dort werden sie von einem generischen Renderer interpretiert und zur Anzeige gebracht. Dateneingaben und Benutzeraktionen werden im Client registriert und zum passenden Zeitpunkt zum Server gesendet, um dort in die Anwendung überführt zu werden.

Dies bedeutet:

- Die gesamte Interaktionslogik der Anwendung findet auf der Java-Server-Seite statt. Eine Entwicklung im Browser-Client (z.B. mit JavaScript) ist nicht notwendig.
- Der Client selbst ist eine im Browser laufende „Rendering Engine“, die vom Server kommende XML-Layouts rendert und entsprechende Eingaben/Ereignisse zum Server zurücksendet.
- Zwischen Client und Server gibt es einen http(s) basierten Datenaustausch. Das Datenvolumen ist äußerst gering, da in beide Richtungen nur Änderungen an Zuständen übertragen werden.

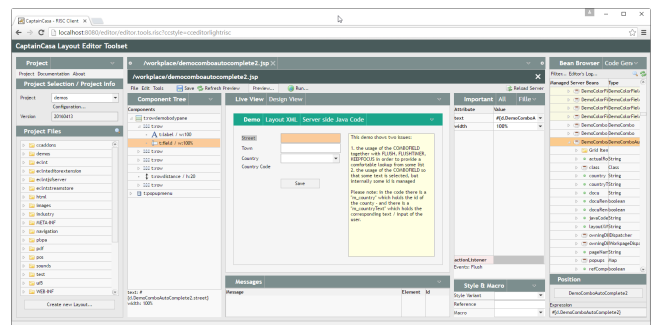


CaptainCasa Enterprise Client RISC beinhaltet eine äußerst reichhaltige, offene Control-Bibliothek, die direkt genutzt werden kann. Das Styling der Controls kann einfach angepasst werden.

Die Bibliothek umfasst neben den „normalen“ Komponenten:

- adaptive Container, in denen der Inhalt abhängig vom verfügbaren Platz angeordnet wird
- komplexe Grids/Trees mit optimiertem Datenaustausch
- Komponenten zur Einbindung externer Frameworks (z.B. OpenStreetMap, Chart.js, SVG, ...)
- Komponenten zur einfachen Darstellung und Pflege von Java- oder Datenbank-Objekten (CRUD-UI, „Scaffolding“)
- Komponenten zur einfachen Umsetzung Rollen-basierter Arbeitsumgebungen

Die Entwicklung von Dialogen auf dem Server wird durch ein umfangreiches Toolset unterstützt, in dem Dialoge per WYSIWYG Layout Editor zusammengestellt werden und an ihre Verarbeitung angeschlossen werden.



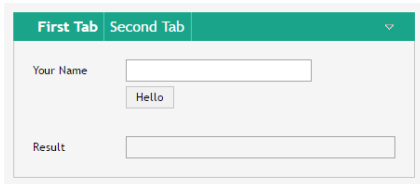
Layout Editor

VORTEILE

Die Vorteile der Nutzung des CaptainCasa Enterprise Client resultieren aus der Client-seitig verwendeten RISC-HTML-Methode einerseits und aus dem Prinzip der Server-seitigen Oberflächenentwicklung andererseits:

- Robuste, langfristig verlässliche, schnelle HTML Clients, die den Anforderungen intensiver, „industrieller“ Nutzung gerecht werden.
- Einfach, effiziente Entwicklung: nur Java auf dem Backend!
- Schneller Einstieg durch klares Entwicklungsmodell und durch hohe Anzahl von bereitgestellten optischen Komponenten.
- Skalierbar und Server-seitig verlässlich durch Verwendung von J2EE Standards. Freie Auswahl der J2EE-Server-Umgebung: Tomcat, Jetty, JBoss, SAP Netweaver, ...

ENTWICKLUNGSMODELL



Auf dem Server werden Dialoge als XML Layout definiert - und entweder statisch hinterlegt oder dynamisch zur Laufzeit erzeugt:

```
<t:rowbodypane>
  <t:row id="g_3">
    <t:tabbedpane width="100%">
      <t:tabbedpanetab padding="20"
        rowdistance="5"
        text="First Tab">
        <t:row>
          <t:label text="Your Name"
            width="100" />
          <t:field
            text="#{DemoHelloWorld.name}"
            width="200" />
          </t:row>
          <t:row>
            <t:coldistance
              width="100" />
            <t:button
              actionListener="#{DemoHelloWorld.onHello}"
              text="Hello" />
            </t:row>
            <t:rowdistance height="20" />
          </t:row>
          <t:label text="Result"
            width="100" />
          <t:field enabled="false"
            text="#{DemoHelloWorld.output}"
            width="100%" />
          </t:row>
        </t:tabbedpanetab>
      <t:tabbedpanetab
        text="Second Tab" />
    </t:tabbedpane>
  </t:row>
</t:rowbodypane>
```

In der Layout Definition werden Attribute von Komponenten entweder direkt gesetzt („100%“) oder sie werden über einen Expression an ein Programm - das sog. Managed Bean - gebunden („#{DemoHelloWorld.name}“).

```
package demo;
import javax.faces.event.ActionEvent;
public class DemoHelloWorld
{
    String m_name;
    String m_output;
    public void setName(String value)
    {
        m_name = value;
    }
    public String getName()
    {
```

```
        return m_name;
    }
    public String getOutput()
    {
        return m_output;
    }
    public void onHello(ActionEvent ae)
    {
        if (m_name == null)
            m_output = "No name set.";
        else
            m_output = "Hello world,
                "+m_name+"!";
    }
}
```

Das „Managed Bean“ bevorratet somit die Properties und Methoden, die von der Oberflächen aus referenziert werden. Das Managed Bean läuft in einer Server-seitigen Session und kann somit direkt auf alle APIs im Server zugreifen.

CAPTAINCASA COMMUNITY

CaptainCasa ist eingebettet in eine offene Community unabhängiger Softwarehersteller aus Deutschland, Österreich, der Schweiz, der Niederlande und Belgien.

Die Community wurde im Jahre 2007 gegründet.

JUMP-START

CaptainCasa steht zum einfachen Ausprobieren per Download zur Verfügung. Für den beschleunigten Einstieg gibt es:

- Screen Sharing Sessions, 30 Min.: „Entwicklung der ersten Seiten“
- „Druckbetankung“ (Workshop, 2 Tage): Sie geben das Szenario vor - und dieses wird mit Ihren Entwicklern umgesetzt. Dabei lernen Sie die Grundlagen des CaptainCasa Enterprise Client kennen - und Sie erhalten einen Prototypen Ihres Szenarios.

LIZENZIERUNG

CaptainCasa Enterprise Client kann komplett kostenfrei genutzt werden! Und zwar in vollem Umfang, für alle Szenarien und ohne Fallstricke!

Kommerzielle Lizenzen umfassen Gewährleistung, einen definierten Service-Level und Investitionssicherheit.

Interesse? Dann freuen wir uns auf Ihre Kontaktaufnahme!

CaptainCasa GmbH
Hindemithweg 13
D- 69245 Bammental
<http://www.CaptainCasa.com>
info@CaptainCasa.com