

OutSystems/CMU MHCI project course for 2008

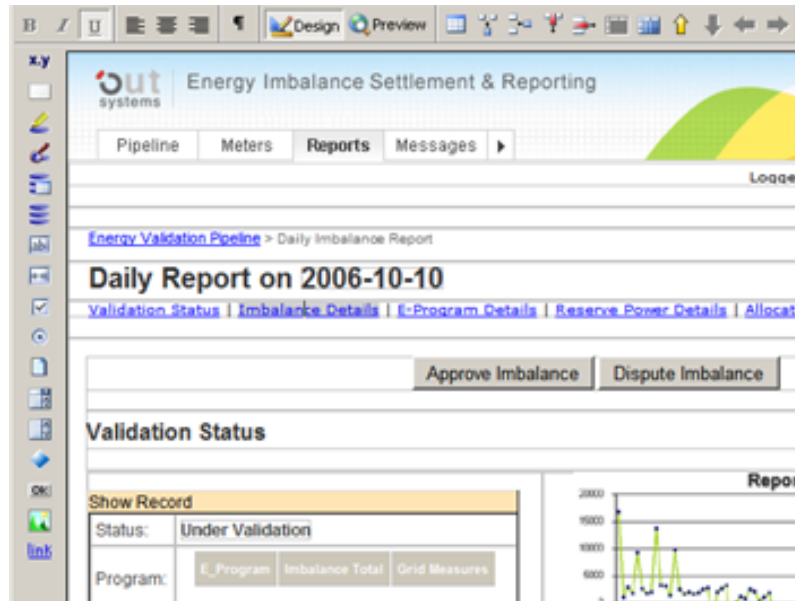
Introduction

The OutSystems Platform is a set of tools and services to support the process of creating and maintaining enterprise applications that integrate with existing systems and reach users through web, email, and mobile user interfaces. At the core of the platform is OutSystems Service Studio, a visual tool that promotes a drag and drop style of composing applications in order to simplify the software development and maintenance process, featuring:

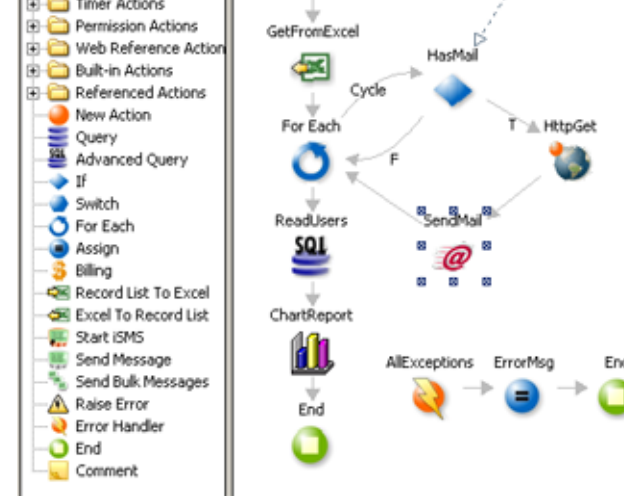
- **All-in-One Application Model** - Service Studio applications includes all the definitions for the user interface, data model, business logic, scheduled processes, web services, and the required integration adapters. This reduces the number of context switches in the development process and simplifies change management as structural definitions are common for all layers;
- **Easy to Get Started** - In a couple of days, developers with basic programming and data modeling skills can feel productive developing basic web and mobile applications. Experienced developers also feel comfortable with the tool as they visually recognize the concepts they already used in the past;
- **Easy Knowledge Transfer** - OutSystems Service Studio promotes a simplified way to compose applications in order to enable anyone within an development team to open, browse, read, understand, modify, and redeploy any application without prior knowledge on the application;

The Human Factor

The software development process is a human intensive process. Simplifying the human interaction and reducing its need and inherent error generation tendency in the software development process, is a key factor to reduce the software development costs. Service Studio uses visual and spacial artifacts to reduce the need of human effort in the recognition and learning processes while developing software and assures a design-time error and consistency check of applications that intend to minimize human errors and detect loose ends or miss-features. The following screen shots give an example of the current visual models used by Service Studio.



User Interface Design



Business Logic Design

Current Challenges

OutSystems Service Studio drove much of the past OutSystems success and image. Nevertheless, a product success requires continuous investment and improvements. The current challenges OutSystems faces are:

1. Service Studio must be simple and attractive to promote word-of-mouth phenomena;
2. The Software development market is very aggressive, forcing OutSystems to have a distinctive approach;
3. Service Studio grew organically in the last years, both in size and complexity, losing some simplicity and coherence;
4. Service Studio denotes usability and productivity issues, both for first time users and experienced users.

Project Goal

The OutSystems/CMU MHCI project goal is to study the changes needed to perform in Service Studio to better support the overall user experience and productivity. To reach this goal, the following Service Studio areas should be visited:

- Image "Look and Feel"
- Layout Organization
- Interaction Paradigm
- Information Visibility

In the following sections we describe the high-level challenges and goals that OutSystems already identified in each area. Nevertheless, the deliverables, the solutions, and the level of detail in each area, is left at the team responsibility.

Image "Look and Feel"

The first area of the project involves reviewing Service Studio icons, colors, and graphic metaphors vs. rational design, in order to provide a distinctive, gratifying, and enjoyable visual experience. The image look and feel must be simple, reduced, and coherent.

Layout Organization

The challenges in this area involve maximizing the space used by the most frequent windows and zones, reducing or removing the less frequent zones whenever possible. This applies to zones like the workspace, the tools tree, the messages window, the operations menu, the project tree, among other specialized editors and windows. It is also relevant to ensure such areas are grouped in order to support common operations that involve multiple areas simultaneously. The resulting layout should help the developers "feel" comfortable with their workspace dimensions and organization for each of the most relevant tasks. Ideally, everything should be a mouse click away in the context of a given task.

Interaction Paradigm

The challenges in this area involve reviewing the developer interaction with the tool in order to:

- Guide new developers for the common operations
- Normalize contextual operations and paradigm
- Normalize navigation paradigm
- Reduce the number of mouse/keyboard context switches
- Reduce mouse distance and clicks
- Support power users

Information Visibility

The last area in this project involves tuning the incremental disclosure approach in order to reduce the number of navigation operations in the knowledge transfer discovery processes. This is critical to support a top-down approach when developers try to read existing code built by someone else. Potential solutions involve pattern recognition, naming, summarizing information, and balancing textual versus graphical information.