

User Interface

The PRPC user interface is the most broadly visible aspect of your application. Its design affects users' productivity, acceptance, accuracy, and satisfaction and so is a critical factor in implementation success. PRPC tools and development techniques ensure that your team will build — quickly and efficiently — a powerful user interface that meets your current and future requirements.

Portals:

Portal: User portals support managers and users of applications as they create, update, route, and resolve work items. A portal is defined by a collection of harness rules that reference a panel set. Each harness rule defines a space, which is the entire portal window. An access group references one or more portal rules.

PRPC includes default portals for the basic 5 operators. We can customize as per our requirement.

The designer used the following to build this form:

Harness: A harness rule defines the appearance and behavior of UI.

Flow Action: A flow action rule dictates how users interact with work item forms to complete assignments. After selecting one flow action, users may fill in a section of the form to complete (perform) the assignment. To define their content, the flow action references a section or sections that contain the form fields the user is to complete.

There are two types of flow actions:

- Connector — Complete the assignment and advance the work object to the next task. The work item is removed from the user's worklist.
- Local — Keep the assignment open and on the current user's worklist.

Sections — Work information and functionality are grouped into sections by context and purpose. Each section can be included in other sections so that the developer can reuse them in a variety of contexts. The designer also uses sections when designing flow actions that require additional fields or forms needed to complete an assignment.

Layouts — Layouts organize properties and controls within a section. Layout cells contain properties, labels, controls, or other sections. There are various types of layouts that dictate how cells are arrayed. The designer used a SmartLayout that ensures consistent spacing of horizontal elements. Layouts can be automatically converted into a sections so that they can be reused in other harnesses, sections, or flow actions.

Controls — Information about a work item is entered or displayed in controls such as text areas, drop-down lists, checkboxes, or calendars. Many of these controls are configured so that they meet, without further refinement, most of your process requirements.

To add a control, the designer simply drags and drops it from a drop-down menu onto a layout cell. The red arrow in the above example indicates how a label was placed in a cell. The designer uses similar menus for adding layouts, containers, and sections to the form.

Repeating Layout: Three types of repeating layouts allow users to view or edit values in a table or tree form. Each type dynamically generates the rows or branches needed to display the requested data.

Grid — The values appear in a spreadsheet format.

Tree — The values appear in a tree structure. The user can click branches to display or hide their leaves. The tree can display embedded relationships as deep as twenty levels.

Tree Grid — The values display in a structure that combines the navigation benefits of a tree with a spreadsheet's access to data.

SmartLayout: A layout grid of paired cells on a harness, section, or flow action form that has columns of uniform width, uniform styles, and uniform characteristics. Each pair of cells holds one label and one property value or other form control. A primary benefit of SmartLayouts is to force vertical alignment even when layouts are nested.

Free Form Layout: : A layout grid of paired cells on a harness, section, or flow action form that holds flexible rows .columns.