```
/****
 *
 * Implementation of add-dialog-card.h
 *
 */

#include "add-card-dialog.h"
#include "card.h"
#include "label.h"
#include "rolodex.h"
#include <InterViews/box.h>
#include <InterViews/button.h>
#include <InterViews/frame.h>
#include <InterViews/glue.h>
#include <InterViews/message.h>


AddCardDialog::AddCardDialog(Rolodex* r, RolodexMenuUI* rmui) : View(NULL, r) {
    /*
     * Construct the add card buttons object.  It contains the OK, Clear, and
     * Cancel buttons.
     */
    acb = new AddCardButtons(r, this, rmui);
}

void AddCardDialog::Compose() {
    /*
     * Make the window of this a VBox, which is an InterViews composite
     * interactor box with vertical layout.  The center alignment is per the
     * specs.
     */
    w = new VBox();
    ((VBox*) w)->Align(Center);

    /*
     * Stick some stretchable glue spacing above the prompting message.
     */
    ((VBox*) w)->Insert(new VGlue(round(0.20 * inch), 0, round(0.50 * inch)));

    /*
     * Insert the prompting message into the top of the vbox.  The prompting
     * message itself is in an hbox with some glue on each side.  This glue
     * will enable horizontal stretching of the message, which stretchability
     * will transfer out to the entire scene that the message is in.
     */
    ((VBox*) w)->Insert(new Label(
        "Enter Information for a Rolodex Card:",
            "times", "bold", "r", 18));

    /*
     * Stick some stretchable glue spacing between the message and the field
     * editors below.
     */
    ((VBox*) w)->Insert(new VGlue(round(0.20 * inch), 0, round(0.50 * inch)));

    /*
     * Allocate the vboxes for the labels and string editors plus the hbox that
     * holds them both.  The outer box forces proper alignment.
     */
    vblabels = new VBox();
    vbseds = new VBox();
    hbouter = new HBox();

    /*
     * Construct a new string editor for each card field.
     */
    ConstructEditors();

    /*
     * Compose the parallel label and string editor vboxes and stick them in a
     * stretchable hbox.
     */
    ComposeLabels();
    ComposeEditors();
    ComposeOuter();

    /*
     * Insert the outer hbox into this' window.
     */
    ((VBox*)w)->Insert(hbouter);

    /*
     * Compose the dialog buttons and insert into the window.
     */
    acb->Compose();
    ((VBox*)w)->Insert(new VGlue(round(0.2 * inch), 0, vfil));
    ((VBox*)w)->Insert(acb->GetWindow());
    ((VBox*)w)->Insert(new VGlue(round(0.2 * inch), 0, vfil));
}

String* AddCardDialog::GetName() {
    return new String(sename->Text());
}

String* AddCardDialog::GetId() {
    return new String(seid->Text());
}

String* AddCardDialog::GetAge() {
    return new String(seage->Text());
}

String* AddCardDialog::GetSex() {
    return new String(sesex->Text());
}

String* AddCardDialog::GetAddr() {
    return new String(seaddr->Text());
}

void AddCardDialog::ConstructEditors() {
    /*
```

```
113       * The constant-length string as the second arg to the StringEditor
114       * constructor is a handy way to set the initial size of the edit box.  The
115       * subsequent call to Message("") clears the blank chars out after the
116       * editor is built and sized.  In this way, the editor properly sized and
117       * initially empty when user frist types in it.
118       */
119      sename = new StringEditor(bs, "                          ");
120      seid = new StringEditor(bs, "                         ");
121      seage = new StringEditor(bs, "                          ");
122      sesex = new StringEditor(bs, "                          ");
123      seaddr = new StringEditor(bs, "                           ");
124
125      sename->Message("");
126      seid->Message("");
127      seage->Message("");
128      sesex->Message("");
129      seaddr->Message("");
130  }
131
132  void AddCardDialog::ComposeLabels() {
133      vblabels->Align(Right);
134      vblabels->Insert(new VGlue(round(0.10*inch), 0, vfil));
135      vblabels->Insert(new Label("Name:  ", "times", "bold", "r", 14));
136      vblabels->Insert(new VGlue(round(0.10*inch), 0, vfil));
137      vblabels->Insert(new Label("Id:  ", "times", "bold", "r", 14));
138      vblabels->Insert(new VGlue(round(0.10*inch), 0, vfil));
139      vblabels->Insert(new Label("Age:  ", "times", "bold", "r", 14));
140      vblabels->Insert(new VGlue(round(0.10*inch), 0, vfil));
141      vblabels->Insert(new Label("Sex:  ", "times", "bold", "r", 14));
142      vblabels->Insert(new VGlue(round(0.10*inch), 0, vfil));
143      vblabels->Insert(new Label("Address:  ", "times", "bold", "r", 14));
144      vblabels->Insert(new VGlue(round(0.10*inch), 0, vfil));
145  }
146
147  void AddCardDialog::ComposeEditors() {
148      /*
149       * The shawdow frames provide the string editor border.  By default a
150       * string editor has no visible border.
151       */
152      vbseds->Align(Right);
153      vbseds->Insert(new VGlue(round(0.10*inch), 0, vfil));
154      vbseds->Insert(new ShadowFrame(sename));
155      vbseds->Insert(new VGlue(round(0.10*inch), 0, vfil));
156      vbseds->Insert(new ShadowFrame(seid));
157      vbseds->Insert(new VGlue(round(0.10*inch), 0, vfil));
158      vbseds->Insert(new ShadowFrame(seage));
159      vbseds->Insert(new VGlue(round(0.10*inch), 0, vfil));
160      vbseds->Insert(new ShadowFrame(sesex));
161      vbseds->Insert(new VGlue(round(0.10*inch), 0, vfil));
162      vbseds->Insert(new ShadowFrame(seaddr));
163      vbseds->Insert(new VGlue(round(0.10*inch), 0, vfil));
164  }
165
166  void AddCardDialog::ComposeOuter() {
167      hbouter->Insert(new HGlue, round(0.25 * inch), hfil, hfil);
168      hbouter->Insert(vblabels);
169      hbouter->Insert(vbseds);
170      hbouter->Insert(new HGlue, round(0.25 * inch), hfil, hfil);
171  }
172
173  void AddCardDialog::Clear() {
174      sename->Message("");
175      seid->Message("");
176      seage->Message("");
177      sesex->Message("");
178      seaddr->Message("");
179  }
```