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

#include "add-card-buttons.h"
#include "add-card-dialog.h"
#include "card.h"
#include "rolodex-menu-ui.h"
#include "rolodex.h"
#include <InterViews/box.h>
#include <InterViews/button.h>
#include <InterViews/frame.h>
#include <InterViews/glue.h>
#ifndef HP700
#include <stdlib.h>
#endif


AddCardButtons::AddCardButtons(Rolodex* r, AddCardDialog* acd,
        RolodexMenuUI* rmui) : View(NULL, r) {
    okab = new OKAddButton(r, acd, rmui);
    clrab = new ClearAddButton(acd);
    canab = new CancelAddButton(rmui);
}

AddCardButtons::~AddCardButtons() {
}

void AddCardButtons::Compose() {
    /*
     * Make the window of this an HBox.
     */
    w = new HBox();

    /*
     * Enter the buttons into the box, delimited with glue.
     */
    ((HBox*)w)->Insert(new HGlue(round(0.2 * inch), hfil, hfil));
    ((HBox*)w)->Insert(okab);
    ((HBox*)w)->Insert(new HGlue(round(0.2 * inch), hfil, hfil));
    ((HBox*)w)->Insert(clrab);
    ((HBox*)w)->Insert(new HGlue(round(0.2 * inch), hfil, hfil));
    ((HBox*)w)->Insert(canab);
    ((HBox*)w)->Insert(new HGlue(round(0.2 * inch), hfil, hfil));
}

OKAddButton::OKAddButton(Rolodex* r, AddCardDialog* acd,
    RolodexMenuUI* rmui) :
        PushButton("OK", new ButtonState(), true) {
                // ^^^   ^^^^^^^^^^^^^^^^^  ^^^^ Default starting state; since
                // ^^^   ^^^^^^^^^^^^^^^^^  ^^^^ we will not use ButtonStates,
                // ^^^   ^^^^^^^^^^^^^^^^^  ^^^^ our starting state is always
                // ^^^   ^^^^^^^^^^^^^^^^^  ^^^^ true
                // ^^^   ^^^^^^^^^^^^^^^^^
                // ^^^   ^^^^^^^^^^^^^^^^^ Dummy value for the ButtonState of
                // ^^^   ^^^^^^^^^^^^^^^^^ this button.  We will not normally
                // ^^^   ^^^^^^^^^^^^^^^^^ use ButtonState objects
                // ^^^
                // ^^^ Text label inside button

    this->r = r;
    this->acd = acd;
    this->rmui = rmui;
}

void OKAddButton::Press() {

    try {
        r->Add(new Card(
            acd->GetName(),                         // Card name as a string
            atoi(acd->GetId()->ConstConvert()),     // Card id as an integer
            atoi(acd->GetAge()->ConstConvert()),    // Card age as an integer
            (Sex)(r->StringToSex(acd->GetSex())),   // Sex as enum literal
            acd->GetAddr()                          // Address as a string
        ));
    }
    catch (AddInputErrors* aie) {
        /*
         * Display zero or more error messages, as appropriate.
         */
        rmui->ShowErrorDialog(aie->GetErrorList());
    }
}

ClearAddButton::ClearAddButton(AddCardDialog* acd) :
        PushButton("Clear", new ButtonState(), true) {
    this->acd = acd;
}

void ClearAddButton::Press() {
    acd->Clear();
}

CancelAddButton::CancelAddButton(RolodexMenuUI* rmui) :
        PushButton("Cancel", new ButtonState(), true) {
    this->rmui = rmui;
}

void CancelAddButton::Press() {
    rmui->ShowEmptyDialog();
}
```