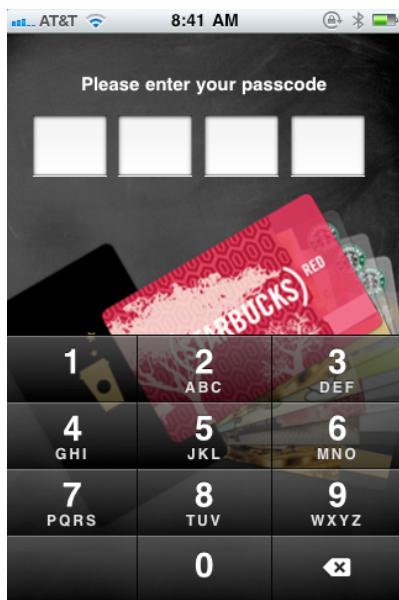


Starbucks

Mobile App Simulator Project Requirements



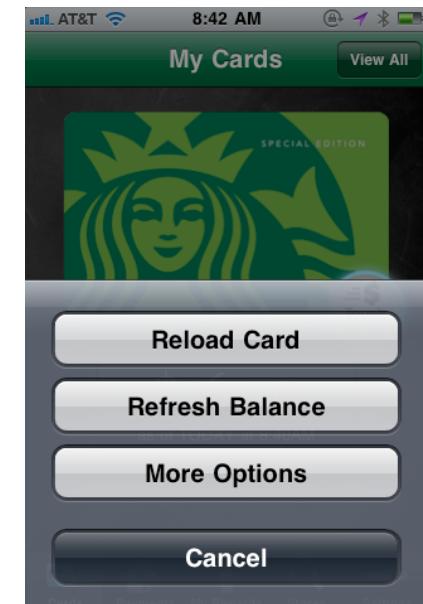
Pin Screen



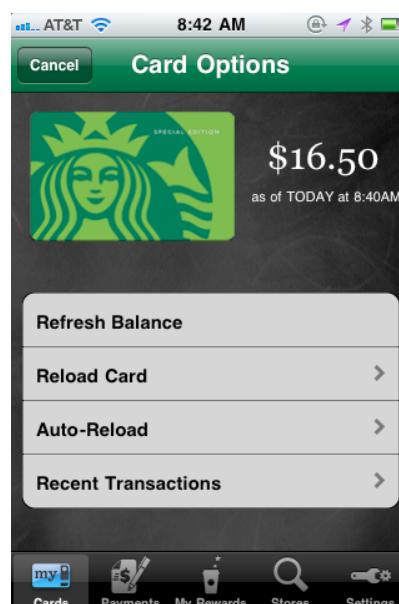
My Cards - Main



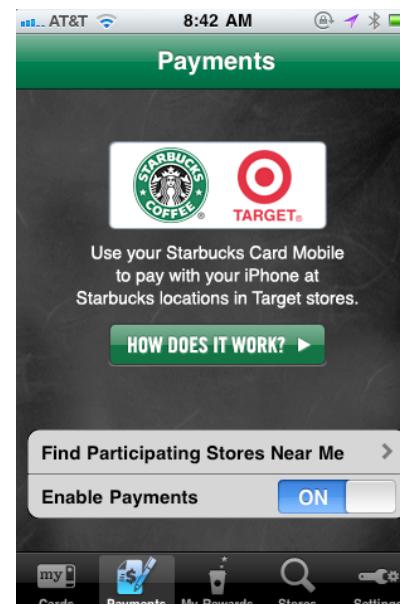
My Cards - Pay



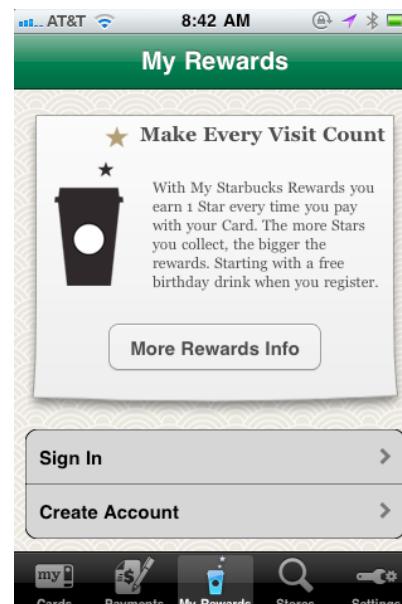
My Cards - Options



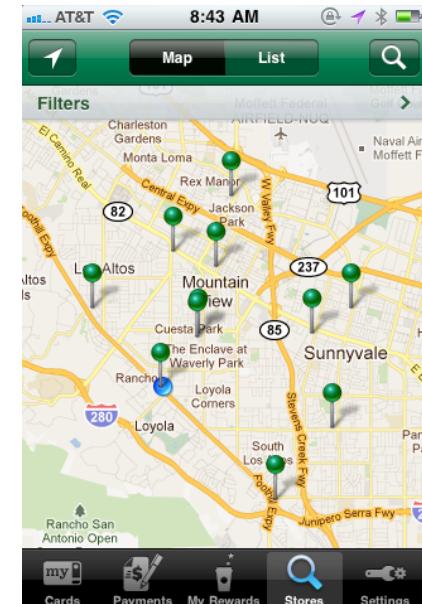
My Cards
More Options



Payment Setup

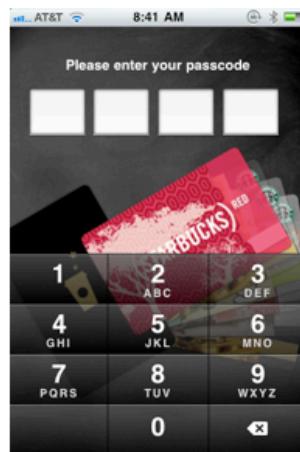


Rewards Setup



Find Starbucks

Solution should implement an “**App Controller**” class which should contain a “display()” method to display the current **Screen Name** as well as well as call the “**display()**” method of the current Screen each time the Screen changes



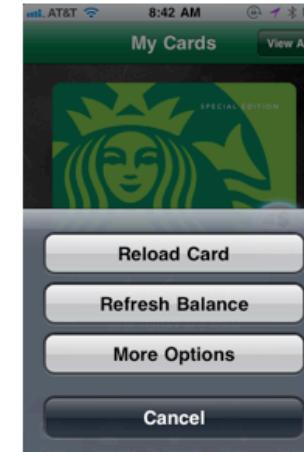
Pin Screen



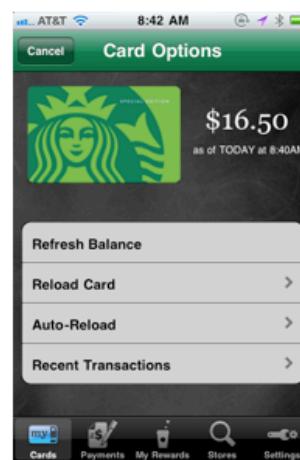
My Cards - Main



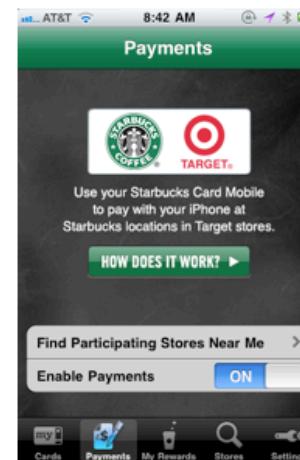
My Cards - Pay



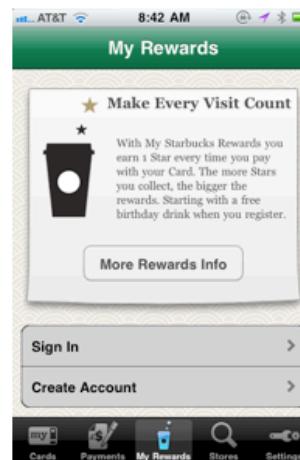
My Cards - Options



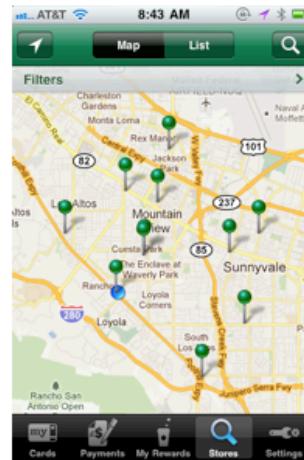
My Cards
More Options



Payment Setup



Rewards Setup



Find Starbucks

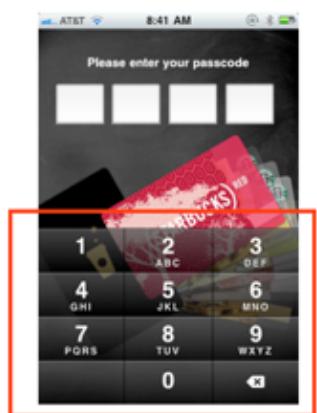
Additionally, ***App Controller*** implements the ***IApp*** Interface, which includes the following operations:

1. **Orientation Operations** (landscape, portrait)
2. **Screen Display** (to render screen contents)
3. **Touch Input** (to send touch events to Screen components)
4. **Execute Command** (to initiate one of five possible commands on menu bar)
5. **Screen Navigation** (next / previous screen)
6. **Screen Name** (to get the Class name of current Screen)
7. **Screen Contents** (to get the on screen content of current Screen)

```
3
4  public interface IApp
5  {
6
7      void landscape() ;           // switch to landscape view
8      void portrait() ;          // switch to portrait view
9      void touch(int x, int y) ;  // send touch event to current screen
10     void display() ;          // display contents of current screen
11     void execute( String c ) ; // trigger a nav bar menu item
12     void prev() ;             // navigate to previous screen
13     void next() ;             // navigate to next screen
14     String screen() ;         // get name of the current screen
15     String screenContents() ; // get contents of current screen
16
17 }
```

All Screens should implement a common interface which includes “***display()***” and “***touch(x,y)***” methods. The “***touch(x,y)***” method accepts the coordinates below. Screens can contain components (such as the “keypad”) which have their own relative coordinates. Touch events can be mapped from the Screen coordinates into the widget’s relative coordinates. For example, given the following layout of the PinScreen, a “***touch(1,5)***” would map into a “***keyPress(1,1)***” on the KeyPad.

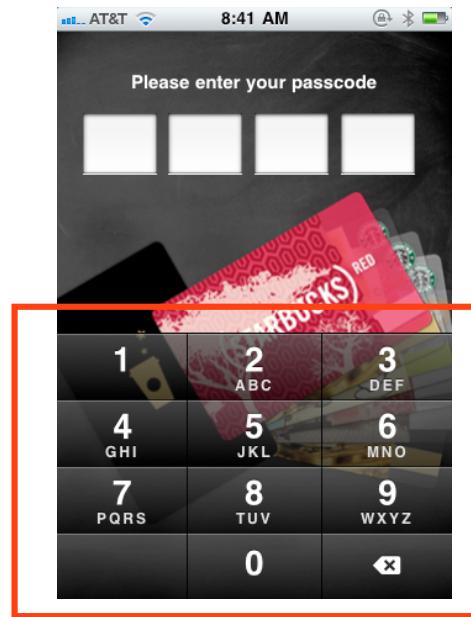
```
5  public interface IScreen
6  {
7
8      void touch(int x, int y) ;           // send touch events to screen
9      String display() ;                 // displays screen components
10     String name() ;                   // returns name of screen
11     void next() ;                     // navigate to next screen
12     void prev() ;                     // navigate to previous screen
13     void setNext(IScreen s, String n) ; // set next screen with action name
14     void setPrev(IScreen s, String n) ; // set previous screen with action name
15
16 }
```



(1,1)	(2,1)	(3,1)
(1,2)	(2,2)	(3,2)
(1,3)	(2,3)	(3,3)
(1,4)	(2,4)	(3,4)
(1,5)	(2,5)	(3,5)
(1,6)	(2,6)	(3,6)
(1,7)	(2,7)	(3,7)
(1,8)	(2,8)	(3,8)

PinScreen Coordinates (x,y)

Pin Screen



(1,1) = 1	(2,1) = 2	(3,1) = 3
(1,2) = 4	(2,2) = 5	(3,2) = 6
(1,3) = 7	(2,3) = 8	(3,3) = 9
(1,4) = _	(2,4) = 0	(3,4) = #

Coordinates: (x,y) = Key

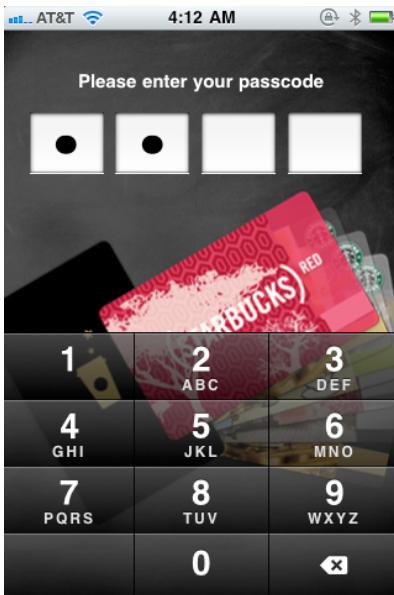
(1,1)	(2,1)	(3,1)
(1,2)	(2,2)	(3,2)
(1,3)	(2,3)	(3,3)
(1,4)	(2,4)	(3,4)
(1,5)	(2,5)	(3,5)
(1,6)	(2,6)	(3,6)
(1,7)	(2,7)	(3,7)
(1,8)	(2,8)	(3,8)

KeyPad
inside PinScreen

Sample Touch Event
Mapping:

=>Touch (1,5)
==>KeyPress(1,1)
==>Key Number 1

Pin Screen



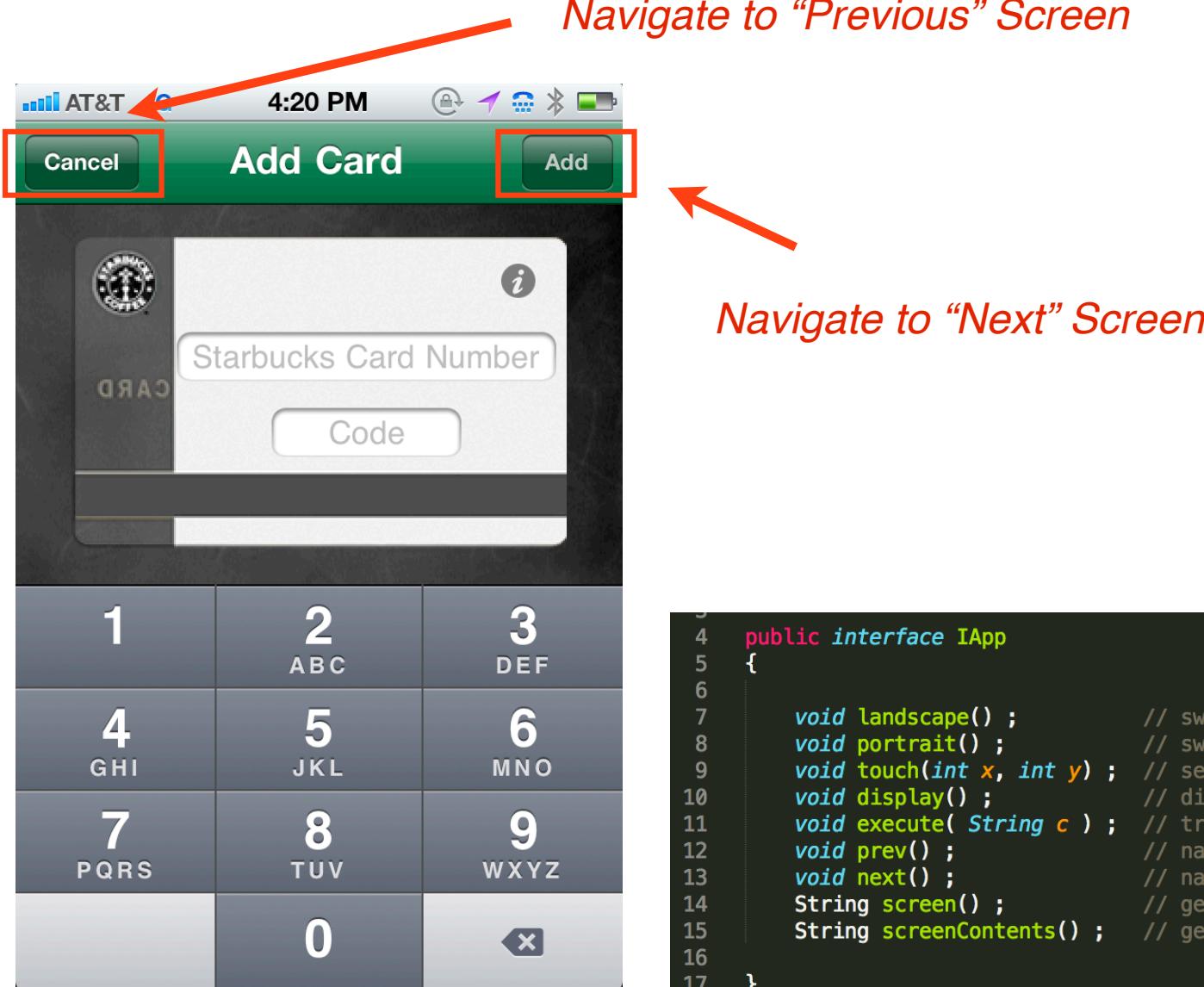
Passcode Display should echo how many pin values were entered. If the “backspace” key was pressed, one pin value is removed.

My Cards - Main



If the valid pin of “**1234**” was entered, the next screen should be the “My Cards - Main” screen showing current Card Balance. If an “Invalid Pin” was entered, then the Passcode Display should display “Invalid Pin”, clear all the pin digits entered and wait for a new pin.

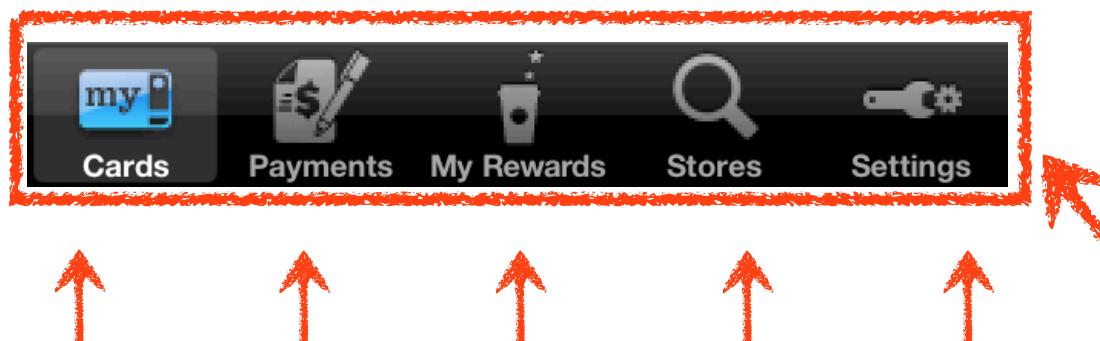
App Controller should implement “**Previous**” and “**Next**” Actions on the **Top Left** and **Top Right** of the Screen — which would be mapped to certain operations in a particular screen (if any at all). For example, the **Add Card Screen maps “Previous” to “Cancel” and “Next” to “Add”**.



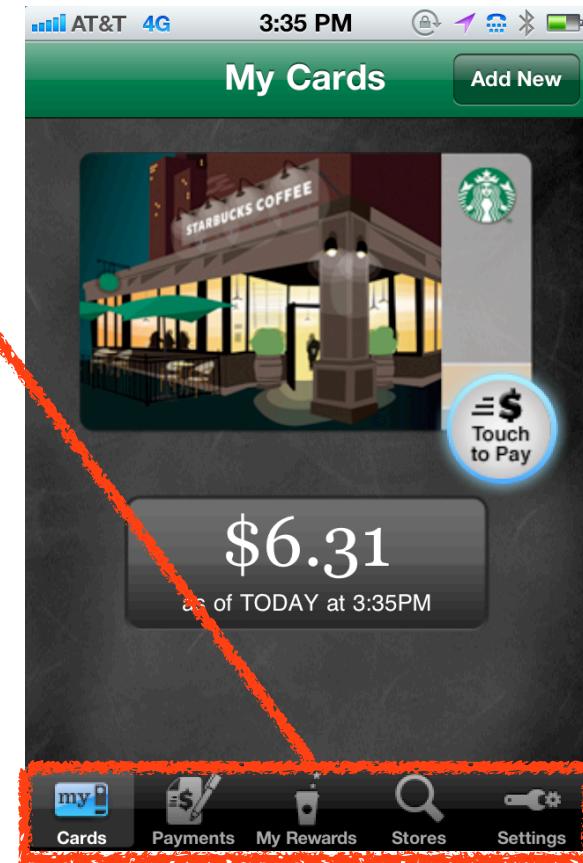
```
3
4 public interface IApp
5 {
6
7     void landscape() ;           // switch to landscape view
8     void portrait() ;          // switch to portrait view
9     void touch(int x, int y) ;  // send touch event to current screen
10    void display() ;           // display contents of current screen
11    void execute( String c ) ; // trigger a nav bar menu item
12    void prev() ;              // navigate to previous screen
13    void next() ;              // navigate to next screen
14    String screen() ;          // get name of the current screen
15    String screenContents() ; // get contents of current screen
16
17 }
```

App Controller also must implement a **MenuBar** (typically position at the bottom of an iPhone Screen) via a **Frame** for managing Screen flows.

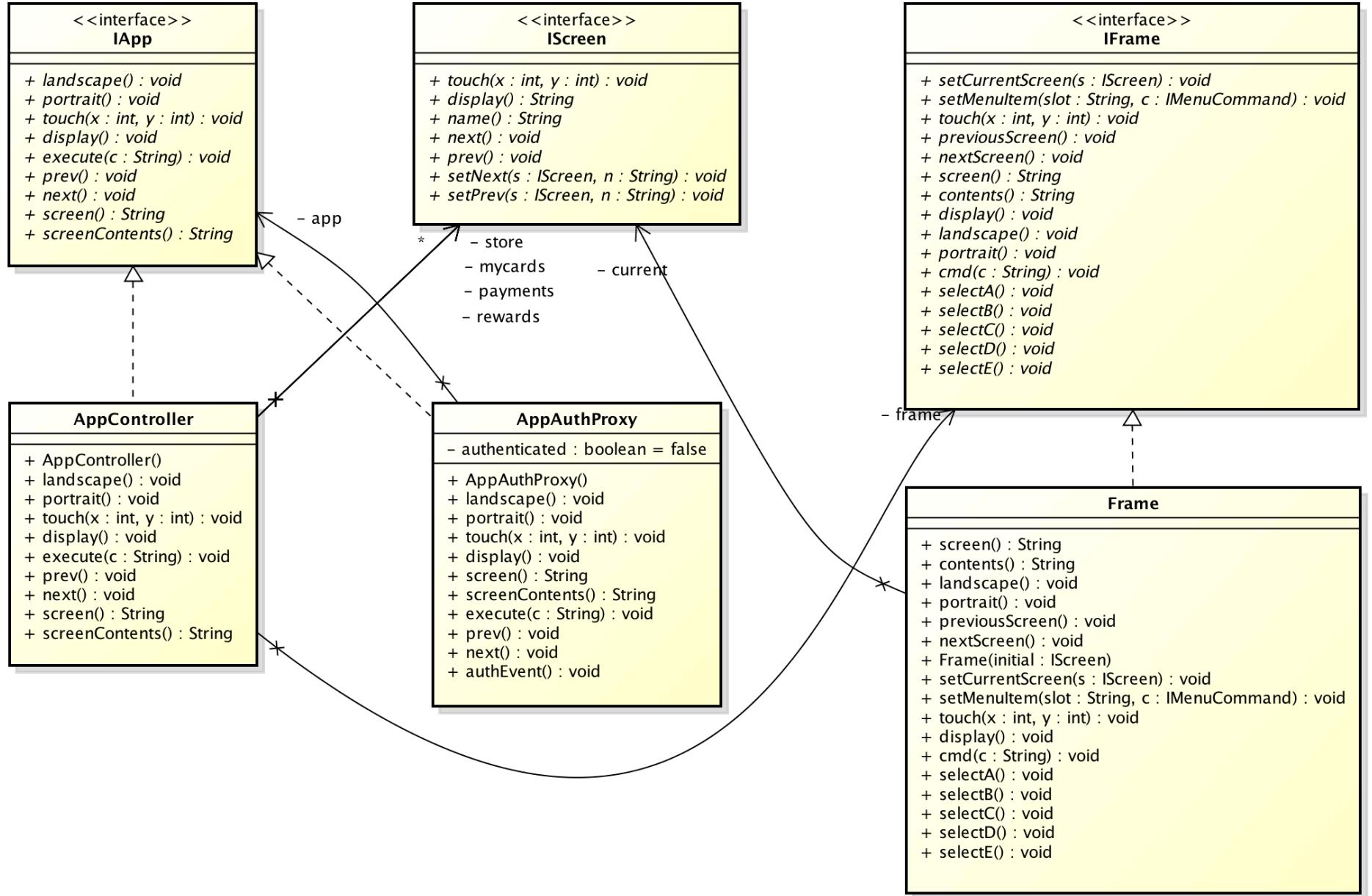
The **IFrame** Interface should include ability invoke any of these **menu items** to represent each of the five menu options an application may have.



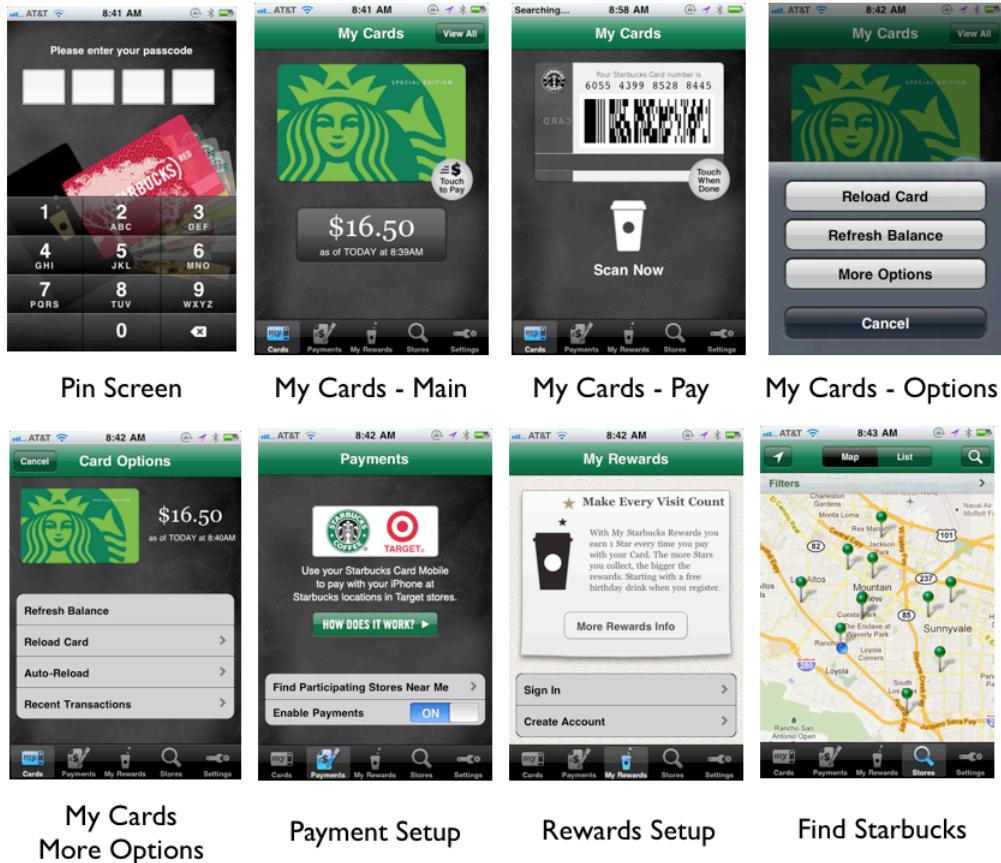
```
3  public interface IFrame
4  {
5
6      void setCurrentScreen( IScreen s ) ;
7      void setMenuItem( String slot, IMenuCommand c ) ;
8      void touch(int x, int y) ;
9      void previousScreen() ;
10     void nextScreen() ;
11     String screen() ;
12     String contents() ;
13     void display() ;
14     void landscape() ;
15     void portrait() ;
16     void cmd( String c ) ;
17     void selectA() ;
18     void selectB() ;
19     void selectC() ;
20     void selectD() ;
21     void selectE() ;
22
23 }
```



UML Diagram of *Interfaces* discussed so far:



Screen Flows



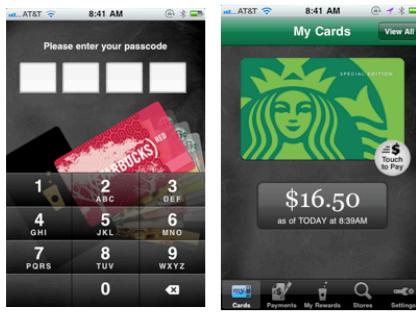
Screen Touch Coordinates

(1,1)	(2,1)	(3,1)
(1,2)	(2,2)	(3,2)
(1,3)	(2,3)	(3,3)
(1,4)	(2,4)	(3,4)
(1,5)	(2,5)	(3,5)
(1,6)	(2,6)	(3,6)
(1,7)	(2,7)	(3,7)
(1,8)	(2,8)	(3,8)

Implement the following Screen Flows and Touch Behaviors:

1. After a “Successful” Pin Validation, the “My Cards – Main” Screen appears
2. A **touch(3,3)** on the “My Cards – Main” screen switches to the “My Cards – Pay” Screen
3. A **touch(2,4)** on the “My Cards – Main” screen switches to the “My Cards – Options” Screen
4. A **touch(1,7)**, **touch(2,7)** or **touch(3,7)** on the “My Cards – Options” Screen
switches to the “My Cards – More Options” screen
5. A **touch(3,3)** on the “My Cards – Pay” screen switches to “My Cards – Main” Screen.

Display Requirements



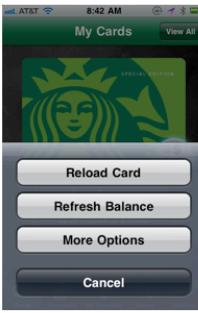
Pin Screen



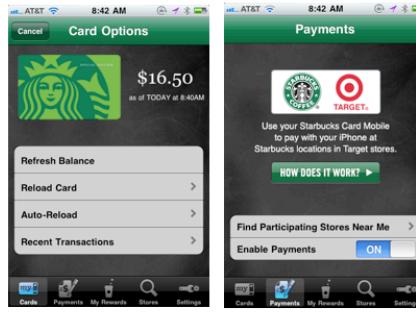
My Cards - Main



My Cards - Pay



My Cards - Options



My Cards
More Options



Payment Setup



Rewards Setup

Find Starbucks

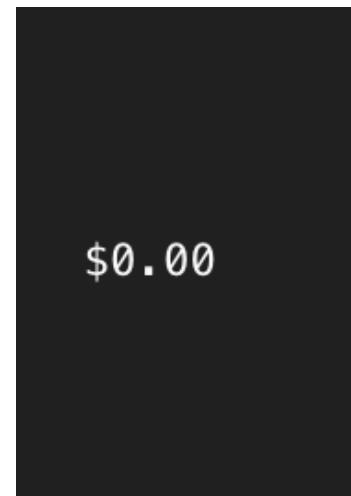
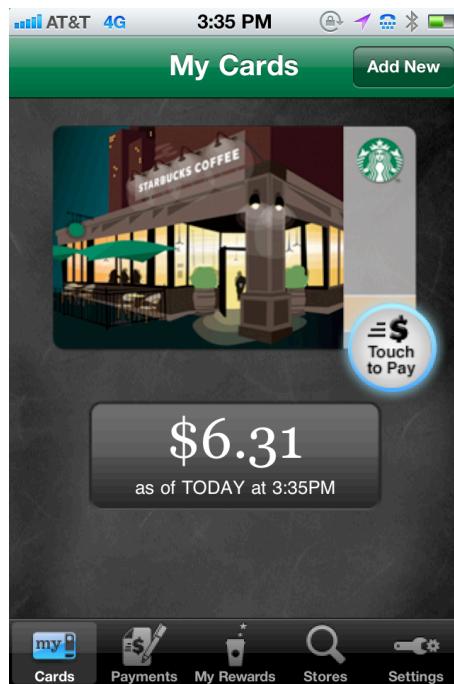
<<interface>>

IScreen

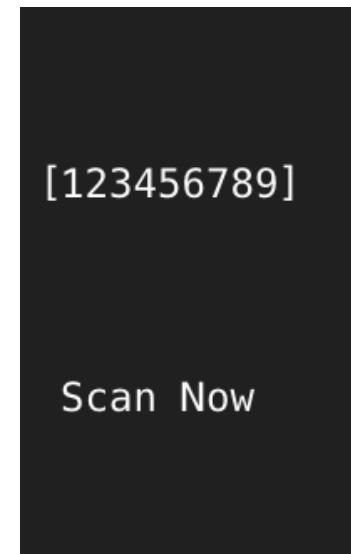
```
+ touch(x : int, y : int) : void
+ display() : String
+ name() : String
+ next() : void
+ prev() : void
+ setNext(s : IScreen, n : String) : void
+ setPrev(s : IScreen, n : String) : void
```

Implement the following Display Output for each screen:

1. Calling “display()” on the “My Cards – Main” Screen should **Show the balance of the current active card.** (Example: \$16.50)
2. Calling “display()” on the “My Cards – Pay” Screen should print-out the words “Scan Now” along with the **Card ID of the active card**.
3. Calling “display()” on the “My Cards – Options” Screen should show the options “Reload, Refresh Balance, or More Options”.
4. Calling “display()” on the “My Cards – More Options” Screen should show the options “Refresh, Reload or View Recent Transactions”
5. Calling “display()” on the “Payment Setup” Screen should show the options “Find Store and Enable Payments”
6. Calling “display()” on the “Rewards Setup” Screen should show the message “Make Every Visit Count”.
7. Calling “display()” on the “Find Starbucks” Screen should show a “Google Map of Local Starbucks”



The “***My Cards - Main***” Screen initially starts out an “empty” card (i.e. Card ID and code of all zeros and a balance of \$0.00). The Screen display, **for example, should show the current Card Balance.**



The “***My Cards - Pay***” Screen should display the **9-digits Card ID** around brackets.

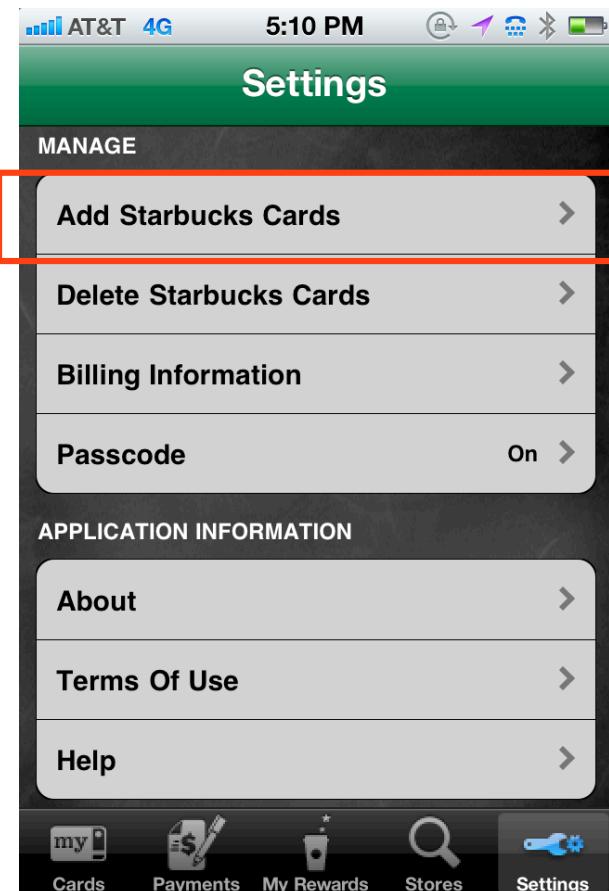
And the text “***Scan Now***”

Settings Screen

To add a Card, a user would touch the “**Settings**” menu and then the “**Add Starbucks Cards**” option. This option has the following touch coordinates: **(1,1), (2,1) or (3,1)**.

Design a **Detailed Sequence Diagram** for this workflow, which should result in creating a new Card object making it the new “Default” card for payment.

Additionally, the “**display()**” method for **Add Card Screen** should print “**Enter a New Card**”; likewise, the “**display()**” method for **Settings Screen** should print “**Manage Card, Billing, Passcode | Show About & Terms.**”



Screen Touch Coordinates

(1,1)	(2,1)	(3,1)
(1,2)	(2,2)	(3,2)
(1,3)	(2,3)	(3,3)
(1,4)	(2,4)	(3,4)
(1,5)	(2,5)	(3,5)
(1,6)	(2,6)	(3,6)
(1,7)	(2,7)	(3,7)
(1,8)	(2,8)	(3,8)

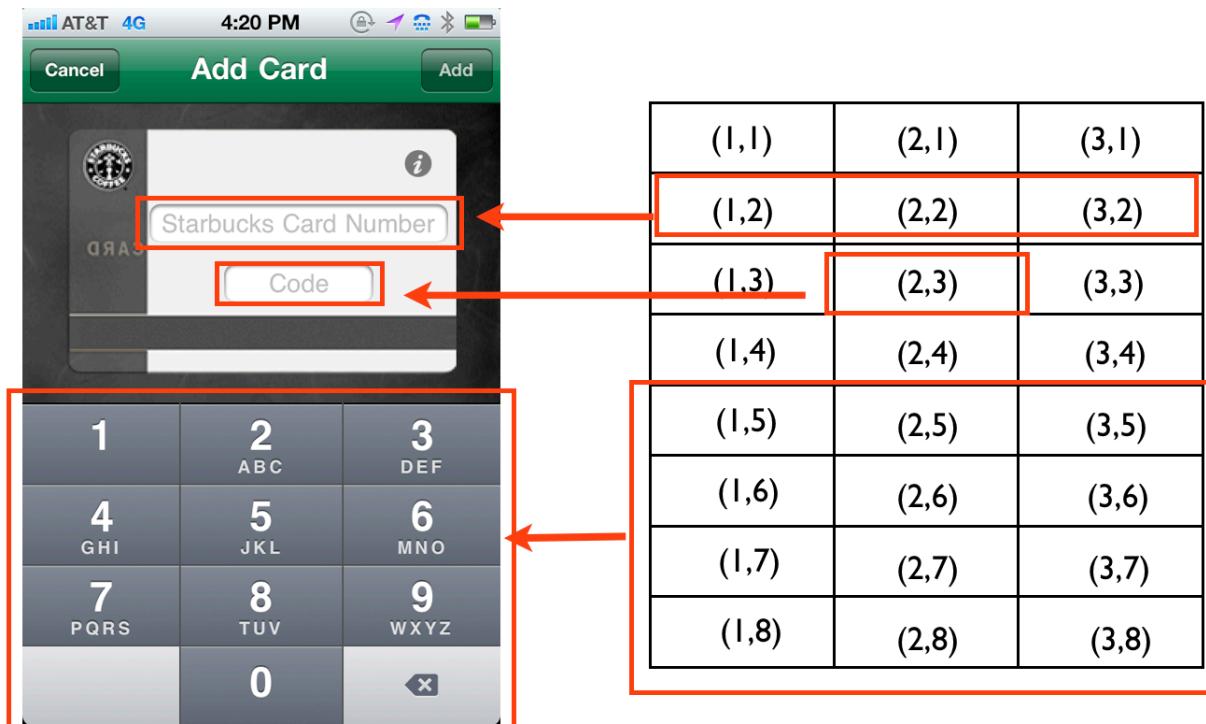
NOTE: Add all new cards with a default initial balance of \$20.00

Add Card Screen - New Card Entry

The “Add Card Screen” should reuse the KeyPad widget. In the “Add Card Screen”, **9 digits can be entered for the Card ID** and **3 digits can be entered for the Card Code**.

A touch() with coordinates of (1,2), (2,2) or (3,2) should set the **Card ID entry as the current focus**. Likewise, touch (2,3) will **switch the focus to the Card Code**. With the focus of the cursor, new digits “touched” should be appended to the current “Card ID” or “Card Code” numbers via the **KeyPad**.

All New Cards will be added with a default balance of **\$20.00** (and will replace any previous Cards entered). That is, the Simulator only manages one card and not a list of Cards. **Only cards with 9 digits card numbers and 3 digits codes are allowed to be added. If this validation fails, clear out both the Card Number Entry Box and the Code Entry Box and wait for the user to reenter.**



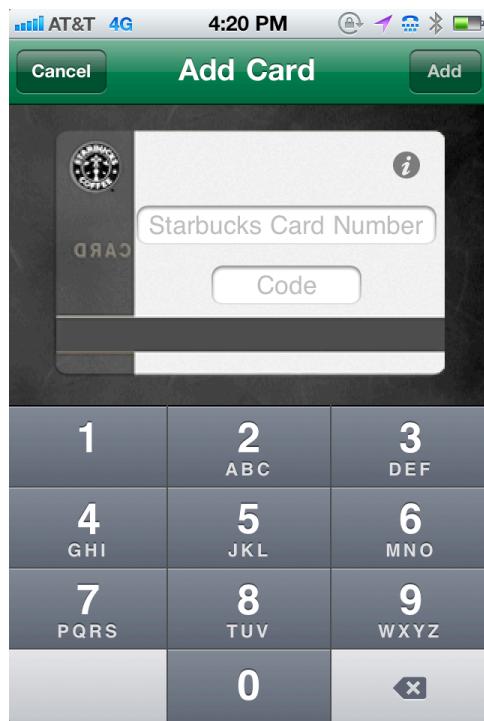
Add Card Screen - Display Format

As Digits are being entered, the “**Add Card**” Screen should Display the following text lines:

The **9 Digits Card ID** around Brackets. I.E. **[123456789]**

The **3 Digits Card Code** around Brackets. I.E. **[123]**

Key Pad Digits



[123456789]
[123]

[1] [2] [3]
[4] [5] [6]
[7] [8] [9]
[_] [0] [X]

**9 Digits
Card ID**

**3 Digits
Card Code**

**Key Pad
Digits**

Add Card Screen - Navigation

The “**Next Nav**” on the Add Card Screen maps to “**Add New Card Request**” with the digits entered and returns to “**My Cards - Main**” showing the new card **balance**.

The “**Prev Nav**” maps to “**Cancel Card Entry**” and returns to the **Previous Screen (i.e. Settings Screen)**.



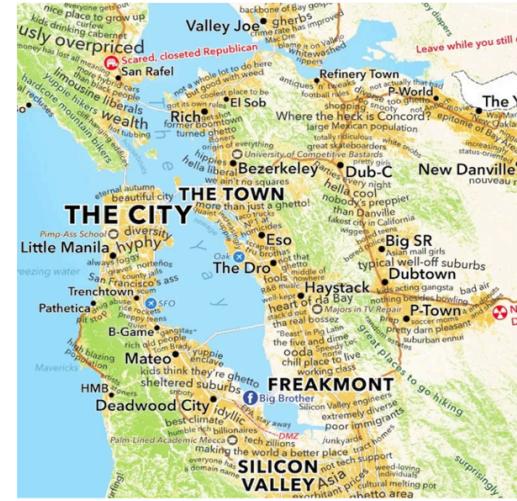
Paying at a Store

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
2															
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4															
5															
6															
7															
8															
9															
0															
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]										

Find Store Screen (Starter Code)

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
2															
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4															
5															
6															
7															
8															
9															
0															
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]										

Find Store Screen (Friendly Screen Name)



Stores:	Touch Cords	Register ID
Dub-C	touch(3,2)	5012349
P-Town	touch(3,7)	1287612
The City	touch(1,3)	6498234
Eso	touch(2,4)	7812386
The Dro	touch(2,5)	8723098
Mateo	touch(1,6)	9621043
Deadwood	touch(1,7)	1393478

(1,1)	(2,1)	(3,1)
(1,2)	(2,2)	(3,2)
(1,3)	(2,3)	(3,3)
(1,4)	(2,4)	(3,4)
(1,5)	(2,5)	(3,5)
(1,6)	(2,6)	(3,6)
(1,7)	(2,7)	(3,7)
(1,8)	(2,8)	(3,8)

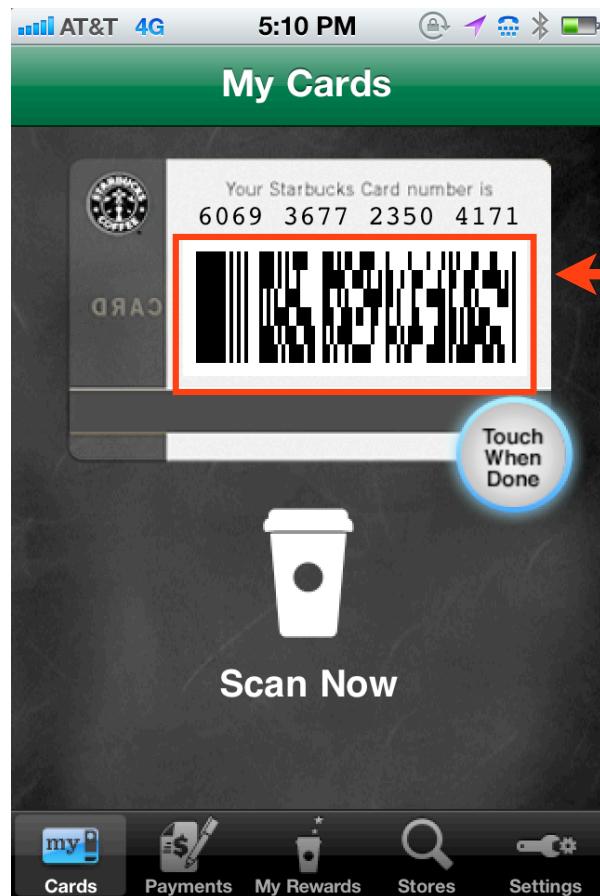
Touching on one of the **Map coordinates for a Store**, select's that Stores' Register ID and then Displays the "**My Cards Pay**" screen for payment.

Making Payments

For Payments, Use the **Starbucks Cash Register App** or **Starbucks API** to place an Order.

The following touch(x,y) coordinates should trigger the payment:

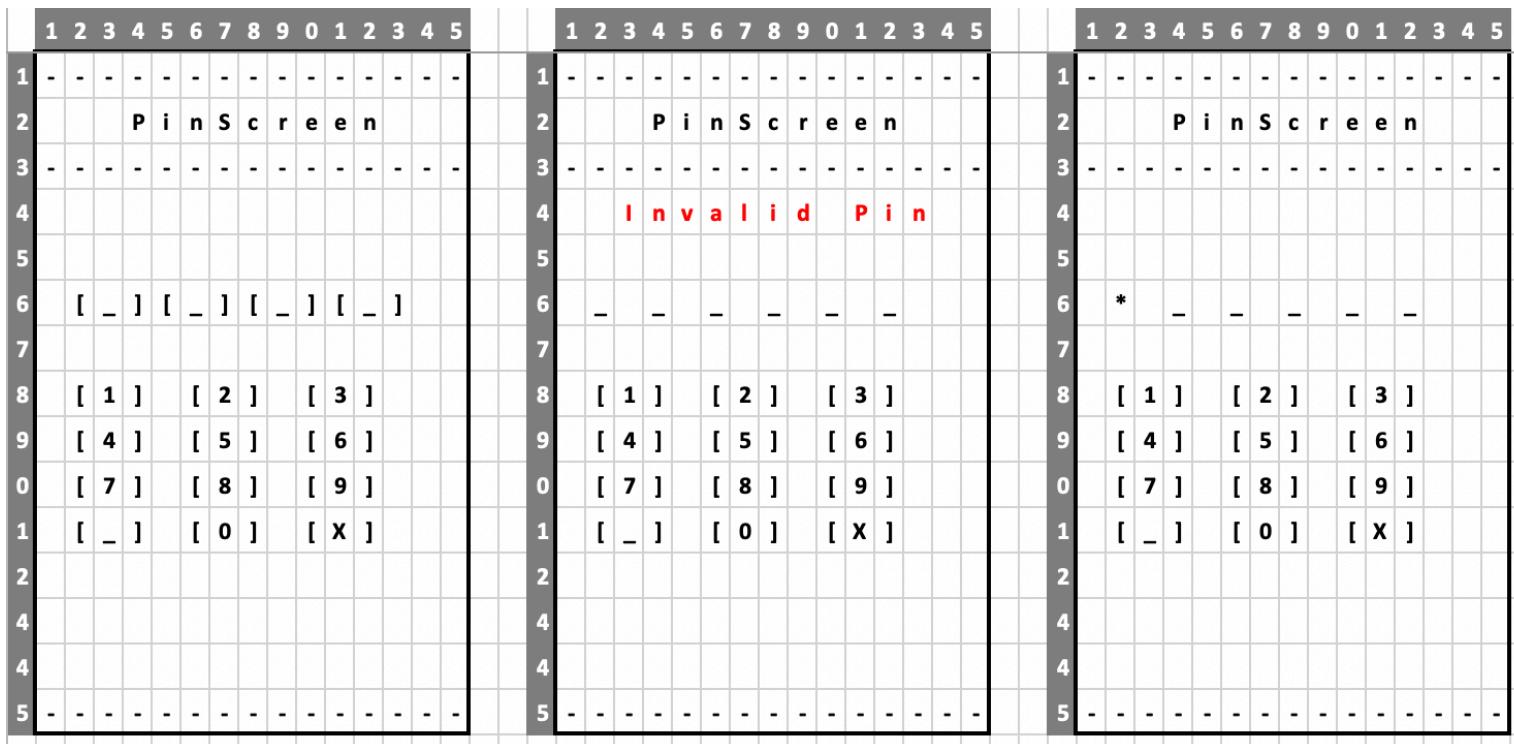
touch (2, 2), touch (3, 2)



(1,1)	(2,1)	(3,1)
(1,2)	(2,2)	(3,2)
(1,3)	(2,3)	(3,3)
(1,4)	(2,4)	(3,4)
(1,5)	(2,5)	(3,5)
(1,6)	(2,6)	(3,6)
(1,7)	(2,7)	(3,7)
(1,8)	(2,8)	(3,8)

Stored Pin Options

The provided “**Starter Code**” for the project defaults to **4-Pin Entry** at Startup with a Pin of “1234”. Review the code provided in Device.java and implement options for starting up the Device with either: **4-pins**, **6-pins** or “**no pins needed**”. For 4-pin or 6-pin options, support the ability to stored a custom pin art startup (i.e. other than “1234”). The **Screen Layout Requirements** are shown in mockups below.



Screen Layout Management

Refactor the design of the Starbucks Mobile App to support a clean code approach to screen layout management in the **Frame Object**.

Existing Frame Object's Contents() Method

```
/**  
 * Return String / Lines for Frame and Screen  
 * @param s [description]  
 * @return [description]  
 */  
public String contents(IScreen s)  
{  
    String out = "" ;  
    out += "=====\\n" ;  
    int nameLen = s.name().length() ;  
    if (nameLen < 14 ) {  
        int pad = (14 - nameLen) / 2 ;  
        out += padSpaces( pad ) ;  
    }  
    out += s.name() + "\\n" ;  
    out += "=====\\n" ;  
    String screen = s.display() + "\\n" ;  
    int cnt1 = countLines( screen ) ;  
    int pad1 = (10 - cnt1) / 2;  
    //System.out.println( "cnt1: " + cnt1 ) ;  
    //System.out.println( "pad1: " + pad1 ) ;  
    out += padLines( pad1 ) ;  
    out += screen ;  
    //dumpLines( out ) ;  
    int cnt2 = countLines( out ) ;  
    int pad2 = 13 - cnt2 ;  
    //System.out.println( "cnt2: " + cnt2 ) ;  
    //System.out.println( "pad2: " + pad2 ) ;  
    //dumpLines( out ) ;  
    String padlines = padLines( pad2 ) ;  
    out += padlines ;  
    out += "=====\\n" ;  
    out += "[A][B][C][D][E]\\n" ;  
    dumpLines( out ) ;  
    return out ;  
}
```

Sample Portrait Layouts



Sample Landscape Layout



Screen Layout Management (Cont.)

Note the following issues:

- The Frame Object has multiple responsibilities: managing the screen layout and the managing the frame. This leads to ugly code in the “**contents()**” method as shown above.
- The Screen layout is currently “**left justified only**”. Changing this to “**centered justified**” would be a painful undertaking. Additionally, it would not be possible to have different screens use different layouts (i.e. left justified for some screens and centered for others).
- Also, the **Landscape Orientation** is not correctly managing Layout. The **Screen dimensions are incorrect, and the Screen Title is not centered**.

Refactor Requirements:

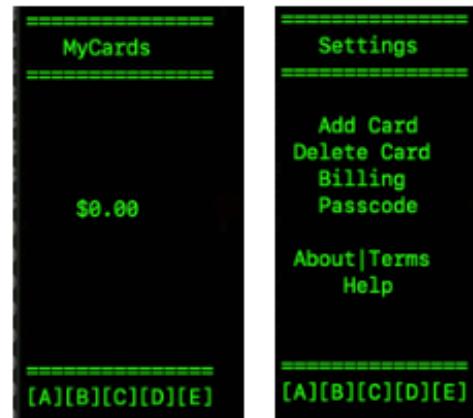
- Refactor the Design for Layout and Orientation Management. **For example, the resulting code for Frame Object’s “Contents()” method should look as shown in the next slide.**
- Support both **Left Justified Layout** and **Centered Justified Layout** on a Screen-by-Screen basis. **All Screens should be “Left Justified” except for “My Cards - Main Screen”, “My Cards - Pay Screen”, “Add Card Screen” and “Settings Screen”, which should be “Center Justified”.**
- Design should also address **Landscape Orientation with proper Screen Size and Layout**. That is, if a Screen is configured for Centered Layout. Screen contents should be “centered” in both Portrait and Landscape mode. **All Screens should default to “Not Support Landscape” except for “My Cards - Main Screen” and “My Cards - Pay Screen”**

Screen Layout Management (Cont.)

Refactored Frame Object's `Contents()` Method

```
/**  
 * Display Screen Contents  
 * @param s Reference to Screen  
 */  
public void display(IScreen s)  
{  
    System.out.println( contents(s) );  
}  
  
/**  
 * Return String / Lines for Frame and Screen  
 * @param s [description]  
 * @return [description]  
 */  
public String contents(IScreen s)  
{  
    String out = "" ;  
    out += "=====\\n" ;  
    out += s.name() ;  
    out += "=====\\n" ;  
    out += s.display() ;  
    out += "=====\\n" ;  
    out += "[A] [B] [C] [D] [E]\\n" ;  
    DeviceConfig.dumpLines( out ) ;  
    return out ;  
}
```

Sample Portrait Layouts



Sample Landscape Layout



Screen “Friendly Names”

Currently, Screen Names are shown as “**Class Names**”. Modify this to use Friendly Names as follows. Note that all the “**Screen Names**” should be displayed as “**Centered Justified**” whether in **Landscape** or **Portrait** Layout Modes.

Class Name Mappings to “Friendly Screen Names”:

- **PinScreen** - Should Display as “**Blank**”
- **MyCards** - Should Display as “**My Cards**”
- **MyCardsPay** - Should Display as “**My Cards**”
- **MyCardsOptions** - Should Display as “**My Cards**”
- **MyCardsMoreOptions** - Should Display as “**My Cards**”
- **Rewards** - Should Display as “**Rewards**”
- **Store** - Should Display as “**Find Store**”
- **Settings** - Should Display as “**Settings**”
- **AddCard** - Should Display as “**Add Card**”
- **Payments** - Should Display as “**Payments**”

Display Requirements (Pin Screens) 15x15 Dimensions & Layout

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	P i n S c r e e n														
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6	[_]	[_]	[_]	[_]	[_]	-	-	-	-	-	-	-	-	-	-
7	[1]	[2]	[3]	-	-	-	-	-	-	-	-	-	-	-	-
8	[4]	[5]	[6]	-	-	-	-	-	-	-	-	-	-	-	-
9	[7]	[8]	[9]	-	-	-	-	-	-	-	-	-	-	-	-
0	[_]	[0]	[X]	-	-	-	-	-	-	-	-	-	-	-	-
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

4-Pin Pin Entry Screen

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6	[*]	[*]	[*]	[_]	-	-	-	-	-	-	-	-	-	-	-
7	[1]	[2]	[3]	-	-	-	-	-	-	-	-	-	-	-	-
8	[4]	[5]	[6]	-	-	-	-	-	-	-	-	-	-	-	-
9	[7]	[8]	[9]	-	-	-	-	-	-	-	-	-	-	-	-
0	[_]	[0]	[X]	-	-	-	-	-	-	-	-	-	-	-	-
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

4-Pin Pin Entry Screen

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6	[*]	[*]	[*]	[*]	[_]	-	-	-	-	-	-	-	-	-	-
7	[1]	[2]	[3]	-	-	-	-	-	-	-	-	-	-	-	-
8	[4]	[5]	[6]	-	-	-	-	-	-	-	-	-	-	-	-
9	[7]	[8]	[9]	-	-	-	-	-	-	-	-	-	-	-	-
0	[_]	[0]	[X]	-	-	-	-	-	-	-	-	-	-	-	-
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

6-Pin Pin Entry Screen

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6	*	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7	[1]	[2]	[3]	-	-	-	-	-	-	-	-	-	-	-	-
8	[4]	[5]	[6]	-	-	-	-	-	-	-	-	-	-	-	-
9	[7]	[8]	[9]	-	-	-	-	-	-	-	-	-	-	-	-
0	[_]	[0]	[X]	-	-	-	-	-	-	-	-	-	-	-	-
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

6-Pin Pin Entry Screen

Display Requirements (My Card Screens)

15x15 Dimensions & Layout

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
2															
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
4															
5															
6															
7															
8	\$	0	.	0	0										
9															
0															
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
5	[A]	[B]	[C]	[D]	[E]

My Cards Main Screen (Starter Code)

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
2															
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
4															
5	\$	0	.	0	0										
6															
7															
8															
9															
0															
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
5	[A]	[B]	[C]	[D]	[E]

My Cards Main Screen (Centered Layout)

Display Requirements (My Card Pay)

15x15 Dimensions & Layout

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
2	M y C a r d s P a y														
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4															
5															
6															
7	[0	0	0	0	0	0	0	0	0]				
8															
9															
0	S	c	a	n											
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]

My Cards Pay Screen (Starter Code)

My Cards Pay Screen (Centered Layout)

Display Requirements (My Card Options)

15x15 Dimensions & Layout

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
2	M	y		C	a	r	d	s		O	p	t	o	n	s
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4															
5															
6															
7	R	e	l	o	a	d									
8	R	e	f	r	e	sh									
9	M	o	r	e		O	p	t	o	n	s				
0	C	a	nc	e	l										
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]

My Cards Options Screen (Starter Code)

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
2															
3															
4															
5															
6															
7	R	e	l	o	a	d									
8	R	e	f	r	e	sh									
9	M	o	r	e		O	p	t	o	n	s				
0	C	a	nc	e	l										
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]

My Cards Options Screen (Centered)

Display Requirements (My Card Options+)

15x15 Dimensions & Layout

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5			
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=			
2	M	y	C	a	r	d	s	M	o	r	e	O	p	t	i	o	n	s
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4																		
5																		
6																		
7	R	e	f	r	esh													
8	R	e	l	o	a	d												
9	A	u	t	o		R	e	l	o	a	d							
0	T	r	a	n	s	a	c	t	i	o	n	s						
1																		
2																		
3																		
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]			

My Cards More Screen (Starter Code)

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5				
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
2																			
3																			
4																			
5																			
6																			
7	R	e	f	r	esh														
8	R	e	l	o	a	d													
9	A	u	t	o		R	e	l	o	a	d								
0	T	r	a	n	s	a	c	t	i	o	n	s							
1																			
2																			
3																			
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
5	[A]	[B]	[C]	[D]	[E]				

My Cards More Screen (Friendly Screen Name)

Display Requirements (Payment & Rewards) 15x15 Dimensions & Layout

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
2	P a y m e n t s														
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4															
5															
6															
7															
8	F	i	n	d		S	t	o	r	e					
9	E	n	a	b		P	a	y	m	e					
0															
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]

Payment Screen

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
2	R e w a r d s														
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4															
5															
6															
7															
8	M	a	k	e		E	v		Y						
9	V	i	s	i	t		C	o	u	n	t				
0															
1															
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]

Rewards Screen

Display Requirements (Find Store) 15x15 Dimensions & Layout

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
2							S	t	o	r	e				
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4															
5							X								
6			X												
7				X											
8					X										
9			X												
0							X								
1				X											
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
5	[A]	[B]	[C]	[D]	[E]										

Find Store Screen (Starter Code)

Find Store Screen (Friendly Screen Name)

Display Requirements (Settings) 15x15 Dimensions & Layout

1 2 3 4 5 6 7 8 9 0 1 2 3 4 5					1 2 3 4 5 6 7 8 9 0 1 2 3 4 5																	
1	=	=	=	=	=	=	=	=	=	1	=	=	=	=	=							
2										2												
3	S	e	t	t	i	n	g	s		3	S	e	t	t	i	n	g	s				
4										4												
5	A	d	d	C	a	r	d			5	A	d	d	C	a	r	d					
6	D	e	l	e	t	e	C	a	r	6	D	e	l	e	t	e	C	a	r	d		
7	B	i	l	l	i	ng				7	B	i	l	l	i	ng						
8	P	a	s	s	c	o	d	e		8	P	a	s	s	c	o	d	e				
9										9												
0	A	b	o	u	t		T	e	r	m	0	A	b	o	u	t		T	e	r	m	
1	H	e	l	p						1	H	e	l	p								
2										2												
3										3												
4	=	=	=	=	=	=	=	=	=	4	=	=	=	=	=	=	=	=	=	=	=	
5	[A]	[B]	[C]	[D]	[E]							

Settings Screen (Starter Code)

Settings Screen (Centered Layout)

Display Requirements (Add Card)

15x15 Dimensions & Layout

1 2 3 4 5 6 7 8 9 0 1 2 3 4 5															
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
2		A	d	d	C	a	r	d							
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
4															
5	[1	2	3	4	5	6	7	8	9]				
6	[1	2	3]										
7															
8	[1]	[2]	[3]						
9	[4]	[5]	[6]						
0	[7]	[8]	[9]						
1	[_]	[0]	[X]						
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
5	[A]	[B]	[C]	[D]	[E]

Add Card Screen (Starter Code)

1 2 3 4 5 6 7 8 9 0 1 2 3 4 5															
1	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
2		A	d	d	C	a	r	d							
3	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
4															
5	[1	2	3	4	5	6	7	8	9]				
6	[1	2	3]										
7															
8	[1]	[2]	[3]						
9	[4]	[5]	[6]						
0	[7]	[8]	[9]						
1	[_]	[0]	[X]						
2															
3															
4	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
5	[A]	[B]	[C]	[D]	[E]

Add Card Screen (Centered Layout)