DZone Refcardz

Application Prototyping
with SketchFlow

By Parvez Ansari

WHY PROTOTYPE?

Application Development

Software applications are an important part of our lives. They help businesses and non-business organizations work more efficiently. Developing an application involves different stages that help shape the application according to client needs.

Prototyping

Just as the foundation of a building is important for its stability, the prototyping phase is a critical part of application development. This Refcard aims to cover the core aspects of application prototyping.

We will also look at how SketchFlow can help us prototype more quickly and efficiently.

WHAT IS PROTOTYPING?

Prototyping helps us create the UI structure of an application without actually writing any code. Problems can be foreseen before they arise, even prior to writing the actual business logic. Prototyping should not be confused with the development of a pilot version of the application.

What does prototyping require? In a real world situation, the client needs to solve a problem. Before you start building the solution it is important to first evaluate the problem to understand exactly what the client needs.

> **Hot Tip**
> The basic goal of application prototyping is to understand the client's needs clearly and get their approval before proceeding with the main application.

It reduces the hassles and costs that might arise while working on the application directly without understanding the requirements.

Prototyping Flow

> **Hot Tip**
> Prototyping is an important component of application development; if the client's requirements are not properly interpreted, you risk wasting time and money.

Let us now see how to prototype an application.

GUIDELINES AND PRINCIPLES

Basic steps to follow for prototyping the application:
- Understanding Requirements
- Documenting the Process
- Technical Feasibility
- Quick Experimentation with Different Ideas
- Client Feedback
- Incorporation of Feedback

Understanding Requirements

When your client comes to you with a problem and tells you what their needs are, it is important that you understand the core requirements so that your end result is what the client wants.

Documenting the Process

This is an essential part of developing the prototype. You need to note down requirements so that you can later measure the prototype's success. The client's feedback from the prototype's review should also be recorded so that you can understand why a certain feature was included or removed from the prototype. This reduces the chance of having a misunderstanding between the client and the developer team.

Technical Feasibility

The client might require a bunch of features that may not be viable to implement on the target computers. So, the

prototype will help judge the technical feasibility of developing the application.

If a required feature is creating a problem in the prototyping stage, it is possible that it might be a bigger roadblock during the actual application development. Thus, prototyping helps the team assess this risk and find an alternate method if necessary.

Vulnerable feature requirements should be tackled first because, if overlooked, they may hinder the project during actual development. Facing difficulties during prototyping makes the actual development process smoother.

Brainstorming
Brainstorming helps the team classify the core components of the project and facilitates the identification of different approaches that can be evaluated to find the most efficient way for interaction between components.

Quickly Experiment with Different Ideas
Prototyping helps us quickly explore how an application should function. Different methods can be experimented with to see how to attain results in the most efficient manner. The main focus is not to work on the business logic, but on the UI and the flow between core components. It helps the team identify a better solution in a limited amount of time. It should be done quickly and the focus should remain on the core requirements.

Client Feedback
Client feedback helps the team polish the prototype and stay focused on the core components. Since feedback is implemented in the initial stages, time is saved.

Improve the Prototype Based on Feedback
Feedback from the client should be incorporated in the prototype. This will help the client identify whether it meets their requirements. The technical team can check to see if it feasible to implement particular aspects of the feedback.

Prototype Should be very close to the Real Application
Care must be taken to ensure that the prototype remains close to what the actual application will be when developed. A feature that can not be implemented in the actual application should not be included in the prototype.

Communication
Just like documentation, communication between the client and the developer team is an important factor for the success of the final application. Communication reduces the risk of a misunderstanding occurring.

REAL WORLD SCENARIO

To better understand the prototype guidelines discussed above, let's consider a real world situation. In the last section of this Refcard we will see how this real world situation is implemented using SketchFlow.

Understanding Requirements
Your client approaches you for the automation of the current manual office system they have in place. The development team should note down the requirements so that they can be used to monitor whether the prototype has met the requirements once it is completed.

Components Involved
The second step is to identify the components of the current manual system. In this situation these components are:
- User Access Rights to different components
- Purchase Order (PO) raising and processing
- Consent Application handling

Login
To ensure authorized access by employees, it is decided to create users based on the roles assigned to them in the office. Proper access privileges are given to users to grant/deny access to a particular component of the system.

Messaging System
The messaging system is the core component of the whole system. A placed purchase order or an application that has been submitted for consent is shown on the respective approving authorities screen. The messaging system also manages the approval/denial of the request.

To ensure decisions are made in a timely manner, a timer that is placed in the messaging system displays the message as soon as the employee logs an application.

The message is displayed in the appropriate employee's message box for them to take necessary action.

Flow Between the components
It is essential to maintain logical flow between the components. To place a purchase order, the flow should be from the main screen to the Purchase Order form and then to the messaging system for approval.

Based on the requirements placed by the client and from the current manual system, the development team decides to create a SketchFlow WPF application.

Client Feedback
Once the prototype is ready, the SketchFlow project is packaged and sent to the client for feedback. The client views the prototype in the SketchFlow player and gives feedback using the "My Feedback" feature of SketchFlow, which is then exported to be sent to the development team. This is then imported into the prototype project so the feedback can be reviewed. Multiple people from the client's team can send different feedback files that all need to be viewed by the development team.

Incorporate Feedback
The next step is to incorporate the feedback into the prototype and then send the prototype back to the client for further review. Several iterations of this can be carried out until the client is satisfied.

Approval of the Prototype
After the prototype is refined with all of the feedback, it has to be approved by the client before the actual development can start.

ABOUT SKETCHFLOW

SketchFlow is a great tool from Microsoft It is part of Expression Studio 4 Ultimate and gives us the capability to quickly and effectively map out and iterate through the flow of an application's UI.

Expression Blend is the production tool for WPF & Silverlight,

whereas SketchFlow is a prototyping tool built on top of Expression Blend. Since it is built on Blend, all of the Blend features are available. This makes our transition from prototype to production much easier.

SketchFlow enables us to quickly present our ideas on how the UI flows and how the screens will look. This enables the client to focus on the functionality rather than the specific details of the design.

Since SketchFlow prototypes are real WPF or Silverlight projects, the prototype or an individual element of the prototype can be reused or used as a resource library.

You can download the 60-Day Free Trial version of Expression Studio 4 Ultimate at http://www.microsoft.com/expression/try-it/Default.aspx

SKETCHFLOW BASICS

Now we will see how to create a SketchFlow application.

Create a Project

Once you start Expression Blend you see the following startup screen. Click on "New Project" to begin.

You get the following screen and have the option to select the project type as Silverlight or WPF. Selecting the project type depends on what kind of requirement the client has, whether the application would be a standalone application or whether it has to be deployed on the web.

In our case our client requires a Desktop application, so we will select **WPF SketchFlow Application**. Provide a suitable project name and click on Ok.
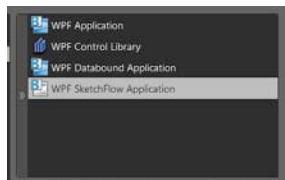
Expression Blend, New Project Screen

You will get the following screen:
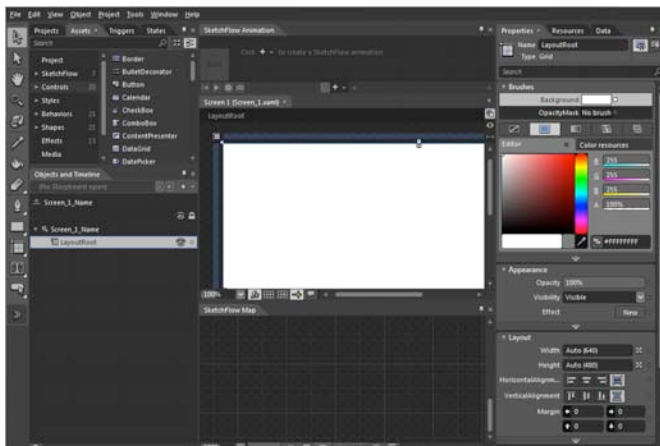
Expression Blend SketchFlow Workspace

If you are familiar with Expression Blend, you will find it similar.You get two views of your design: Artboard and SketchFlow Map.

Artboard shows us the content of the individual screens of our prototype. This is the visual part of the application where you can use the full range of controls available in Expression Blend. You can draw by using the drawing tools. You can also import images from Adobe Photoshop, which is especially useful if the client has a visual mockup of the application he would like to use.

SketchFlow Map appears at the bottom of the Expression Blend application when working in SketchFlow. It is where the whole flow of the application is created. It allows us to define the way each screen interacts with the other screens.

In the below figure you can see that the flow of the application is from Screen1 to Screen2. So, it can be determinted that Screen1 is the startup screen.
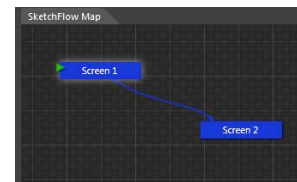
SketchFlow Map

are an important part of the prototype because they are the visual element visible to the user. Each screen is shown as a node in the SketchFlow Map.

There are two types of screens/nodes:
- Normal screens
- Components screens

Normal screens are generally used to navigate to and from. The component screen is what can be reused in multiple screens, for example a menu component created by the design team can be reused on multiple screens as required.

When you hover over the node in the SketchFlow Map panel a visual menu appears, as shown below. You can also see the XAML file name of the screen in the tooltip

Screen Node

Hot Tip — To create a new screen click on the Create a screen menu in the SketchFlow Map toolbar of the node selected

If you want to connect to the existing screen, hover on the selected node and from the visual menu select the "Connect an existing screen" icon. Start dragging to create a new navigation connection. An arrow will appear following the mouse pointer. This arrow has to be dragged to the screen where you want to make the connection.

Hot Tip — To remove the flow map connection right click the connection and click remove.

Projects Panel allows us to view, edit, and manage all files associated with the open project.

Objects and Timeline panel shows the hierarchical view of the contents on the screen. It allows us to select objects and move them around as required.

Layers can be further added to organize the objects on the artboard. When you create a storyboard a red circle indicates that recording is on. You can select the storyboard from the Storyboard Picker.

The image below (a) indicates the Record Indicator, (b) indicates the Storyboard Picker and (c) represents playback controls.
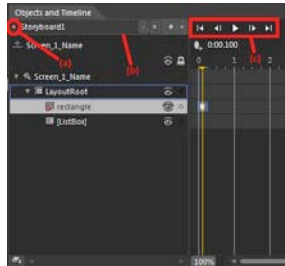
Objects and Timeline Panel

Properties Panel is an important UI part of SketchFlow that you will use often. It shows the properties of the currently selected item on the artboard. Properties like Background, Name etc. can be changed for a particular component of the application.

If an object is modified in the artboard directly using object handles, the property changes will be reflected in the properties panel. The sections of the properties panel are collapsible.

Important sections of the properties panel are (these may vary according to the object on the artboard selected): Brushes, Appearance, Layout, Common Properties, Transform, Miscellaneous.

> **Hot Tip**
>
> There might be confusion on what is the most effective way to change the property of an object on the artboard or in the properties panel. It depends on the requirement. If you wish to fit in an object on the artboard it would be better if you use the object handle and mouse to arrange it on the artboard.
>
> If you want a specific height and width of a control you can do it by selecting the object on the artboard and modifying the properties in the properties panel.

The changes you make in the object properties are registered in the form of XAML code for example if you change the height and width of the Rectangle to 100 and 200 respectively, the XAML code will look like this:

```
<Rectangle x:Name="rectangle" Fill="#FFF4F4F5"
HorizontalAlignment="Left" Height="200" Margin="74,93,0,0"
Stroke="Black" VerticalAlignment="Top" Width="100"/>
```

Other properties are set by Blend itself based on the object placement you made like **VerticalAligment** and **Margin**.

USEFUL KEYBOARD SHORTCUTS

The following shortcut keys are the most frequently used keys in Expression Blend.

Show/Hide SketchFlow Map	Shift + F12
Hide/Show all panels	Tab, F4
Make into Component Screen	Shift + F8

Open the Assets Panel	Ctrl + Period
Toggle between Design, Split View and Code View	F11
Build the Project	Ctrl+Shift+B
Run the Project	F5

USING SKETCHFLOW TO PROTOTYPE

SketchFlow is a great tool for prototyping the application. We have already seen the components of the SketchFlow Application and how to create a new project. We will now start to prototype the application.

Start a new SketchFlow WPF project. By default SketchFlow starts with many panels open. We may not be using all of these panels so it is recommended to keep only the following panels open: Artboard, SketchFlowMap Panel, Properties Panel, Projects Panel, States Panel.

If you require other panels later you can always open them using the Windows Menu.

> **Hot Tip**
>
> If you get to a point where you have made mistakes on all of the open panels and are confused, you can always revert back to the SketchFlow's default workspace by pressing Ctrl+Shift+R or by selecting "Reset Current Workspace" from the Windows Menu.

Creating Screens

In the SketchFlow Map Pane you will see a node named Screen1. Right click on it and rename it as "Welcome Screen."

Double click on the Welcome Screen node, this will select the Welcome screen page in the artboard and allow you to draw the contents you want in the welcome screen.

Hover over the welcome screen node and click on the "Create a connected screen" node. This will create a new node, name it "Login". Double click on login to select the "Login Screen".

Select the "Objects and Timeline Panel" and click on the root component, in this case it is UserControl named "Login_Name". In the properties panel enter Height and Width as 300 px. This will resize the Login Panel.

From the left side tools panel click on the Assets button ⟫ it will open the following panel
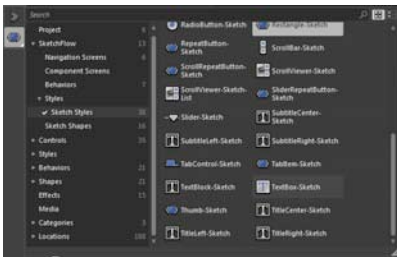
Assets Popup

From under the Sketch Styles select TextBox Sketch to draw the TextBox. The Login Screen will look like this.
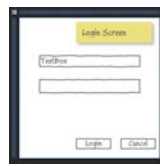
Login Screen

Just as we created the Login Node from the Welcome Node, create the "Main Screen" node from the Login Node. Double click on the new node to select the screen.

From the "Main Screen" node create three more screen nodes and name them: Consent Application Form, Purchase Order Form, Help, Messaging System.

Now, the basic structure for our prototype is ready.

A navigation menu has to be created which will allow us to move around the screens.

Double click on the "Main Screen" node to select it. On the artboard create a rectangle and three buttons as shown below:

Navigation Component Screen

In the Objects & Timeline panel select the three buttons and rectangle that we have created, right click on the selection and click on the "Make Into Component Screen" option as shown below.

Enter the name of the new Component Screen as "Menu", Blend will create a separate screen with the items you had selected. On the SketchFlow map, the Component Screen node is shown in green.

When you hover over the component node in SketchFlow Map you get a similar menu as the normal screen node. This component screen we just created is to be shown in the consent Application, Purchase Order & Help screen.

Drag the green component screen node over the consent form to link it. Once the link is established, the menu element will be visible on the consent form screen. Similarly, drag the green node over the PO and Help nodes.

The messaging system will pass on the approval or denial of the request to the main screen of the user. So, a flow from the messaging system to the Main screen has to be created. From the messaging system menu drag the option "connect an existing screen" to the main screen.
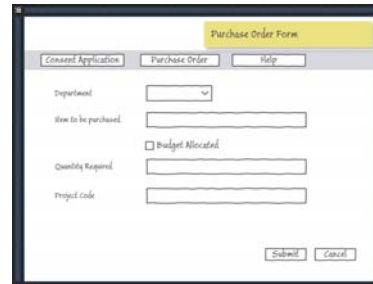

SketchFlow Map

Creating Contents in the Screens

Create content in the MainScreen, Consent Application, PO, Help and Messaging System. The dummy content on each screen gives the client a rough idea of how the application will look.
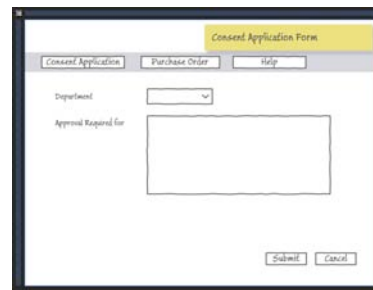
Consent Application Form

Purchase Order Form

Main Screen

Creating Flow Between the Components

Select the welcome screen. In the Objects & Timeline Panel, right click on the root element "Layout Root" and under the option "Navigate to" select "Login Screen".

Select the Login Screen and create a navigation Link on the "Login" button to the "Main Screen". Similarly navigation flow can be created for all the remaining screens.

Under the "Project Panel" double click and select the Component Screen "Menu", Right click on the "Consent Application" Button to go to the Consent Application Form, similarly click on the PO Button and the Help button to go to respective screens.

> **Hot Tip**
> Since the Menu is a Component Screen the navigation flow will apply in the screens where it is used.

The navigation of the "Submit" button in the Consent & PO Screens is linked to the Messaging Form and the "Cancel" button is linked to the Main Screen.

Sending the Prototype for Feedback

After the completion of the Prototype, we send it to the client for his comments. Under the File Menu click on the "Package SketchFlow Project" option. Enter the Folder location where you want the package to be exported. The packaged project is sent to the client for review.

When the client executes the application sent to them they see the "SketchFlow Player"
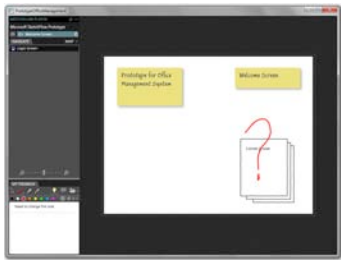
SketchFlow Player

The player allows the client to navigate through the application. They can provide their response using the feedback panel. The ink feedback allows drawing over the screen, and respective text feedback can be typed in the feedback panel. To send feedback, click on "Feedback Options Button" .

From the opened menu select "Export Feedback". It will ask for the Author information and location to save the feedback file to. Save the feedback file (client will send this file to developer team).
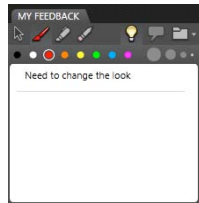

Reviewing the feedback
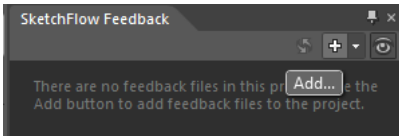
Once the developer team receives the feedback file, it can be loaded into the prototype project. Open the "SketchFlow Feedback Panel". If it is not open click on the "SketchFlow Feedback" option under the "Windows" Menu.

Click on the Add button to load the feedback file sent by the client. Once the feedback file is loaded you can see the details in the Panel.
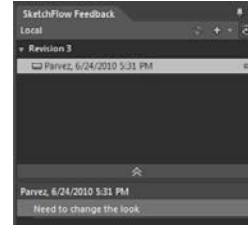
SketchFlowMap will show a small yellow bulb over the node on which the feedback is given

Node with Feedback

The developer team can watch the ink input by clicking on the eye icon in the SketchFlow Feedback Panel.

Multiple feedbacks from different members of Client Team can be loaded together.

The developer team incorporates the feedback from the design team and sends a new export of the SketchFlow player. Client reviews the new build. Once this iteration is complete the developer team is ready to start work on the Pilot Project.

ABOUT THE AUTHOR

Parvez Ansari is the Chief Architect of Nukeation (www.nukeation.com) a leading UX studio. He has been working on WPF since before its public release. He is part of the Reuxables Team who are worldwide pioneers in XAML themes and have been responsible for the official Microsoft Themes for WPF and Silverlight.

RECOMMENDED BOOKS

This book is for designers, user experience pros, creative directors, developers, or anyone who wants to create rich, interactive, and compelling products. If you want to communicate innovative ideas, research, experiment, and prototype in the language of the interface, Dynamic Prototyping with SketchFlow in Expression Blend is the perfect text. Learn how to sketch, iterate, and validate ideas—utilizing the power and productivity within SketchFlow.

BUY NOW
books.dzone.com/books/dynamic-prototyping

Browse our collection of over 100 Free Cheat Sheets

Getting Started with Cloud Computing
By Daniel Rubio

Core HTML
By Andy Harris

Free PDF

Upcoming Refcardz
Network Security
Hadoop
UML
Subversion

DZone

DZone communities deliver over 6 million pages each month to more than 3.3 million software developers, architects and decision makers. DZone offers something for everyone, including news, tutorials, cheat sheets, blogs, feature articles, source code and more. **"DZone is a developer's dream,"** says PC Magazine.

ISBN-13: 978-1-934238-75-2
ISBN-10: 1-934238-75-9

50795

9 781934 238752

$7.95

Version 1.0