Wireframing for user workflows
Treat wireframes as state machines

- Can help reveal poor page flow
  - Dead ends
  - Overly complicated paths
  - Error message handling
Wireframing a series of HTTP requests

- **incoming arc** points to **starting state**
- **rounded rectangles** represent different **views (states)**
- **labeled arrows** represent **state transitions**
- **arrow labels** represent **HTTP requests, transition trigger actions**
Logging in & out of Facebook (with 2-factor authentication)

GET facebook.com

- login screen
  - enter correct login details
  - enter incorrect login details

- security code screen
  - if new or not-remembered browser
    - ask to resend code
  - if remembered browser
    - enter incorrect security code

- news feed
Logging in & out of Facebook (with 2-factor authentication)

1. GET facebook.com
2. login screen
   - enter correct login details
3. security code screen
   - enter correct security code
   - ask to resend code
4. "remember browser?" screen
   - if remembered browser
     - click profile photos, names
     - click profile photos, names
   - if new or not-remembered browser
     - enter incorrect login details
5. news feed
   - click profile photos, names
   - click "home" link
6. user profile
   - log out
7. respond "yes"
8. respond "no"
Logging in & out of Facebook (with 2-factor authentication)

GET facebook.com

**login screen**
- enter correct login details
- enter incorrect login details

if new or not-remembered browser
- if remembered browser
- ask to resend code

**security code screen**
- enter correct security code
- enter incorrect security code

**“remember browser?” screen**
- respond “yes”
- respond “no”

**user profile**
- click profile photos, names
- log out

**news feed**
- click profile photos, names
- click “home” link
- log out

cleaned-up state machine diagram
What can we learn from this wireframe?

- It takes 3 pages to log in for new browsers! 2-factor auth **purposefully** adds friction for security.

- Facebook presents the option of “remembering browsers” if the user wants some added security, but not too much friction.
Wireframing AJAXed/client-side interactions

- Use a large rectangle for the view.

  - Label how you enter the initial sub-state.

  - Use nested rectangles for sub-states.

  - Sub-state transition.

  - Sub-state 2.

  - State transition.

  - View 2.
Posting a new status on Facebook

1. **click the status input form**
2. **tag friend**
3. **untag friend**
4. **add photo**
5. **remove photo**
6. **add hashtag**
7. **remove hashtag**
8. **click “post status” button**

**NOTE:** The sub-state transitions are local updates happening on the client. The state transition (clicking “post status”) sends the entire update to the server.
What can we learn from this wireframe?

Facebook wants everyone to stay on the News Feed, so the page supports both read and write.

Not the entire new status form is visible when you’re just browsing. When a user expresses an interest in updating his status, he’ll click the box.

Clicks (and hovers) are ways for users to express interest. **Reduce friction by reducing # clicks needed to get something done.**
Wireframing for page layout
Wireframing individual pages

• Separate page structure from aesthetics
• Focus on position & hierarchy
• No colors or images
1. Determine layout with boxes

- Logo
- Search bar
- Navigation
- User profile links
- Favorites
- Pages
- Groups
- Apps
- Chat
- Update status form
- News feed
- Ads
- Birthdays
- Fine print
2. Add information hierarchy
3. Add grayscale coloring
What can we learn from this wireframe?

Do you ever use these links on the left?

Why does a post’s details have a colored background?
What can we learn from this wireframe?

- **Why does a post’s details have a colored background?**
  - Groups details together. User will treat it as 1 unit when skimming.
  - Actions on a post are *not* on a colored background - they are designed to stand out.

- **Do you ever use the links on the left?**
  - In New Timeline, the entire left column is gone and placed in a drawer menu instead.

- **Find the balance between a flat page structure and not overwhelming the user with options.**
Wireframing tools

• Pencil & paper!

• **Keynote & PowerPoint**
  All wireframes here were made in Keynote.

• **Balsamiq** [balsamiq.com](http://balsamiq.com)
  Software for wireframes.

• **OmniGraffle** [omnigroup.com/omnigraffle](http://omnigroup.com/omnigraffle)
  Software for wireframes.
Wireframing tools

• **Lucidchart** lucidchart.com
  Software for flowcharts & wireframes.

• **Wirify** wirify.com
  Browser extension that lets you see a “wireframed version” of a website, to show you the difference between a wireframe and the final implementation.