Design

Knowing what you want to build and why

Prerequisites

- None

User Experience Tools

- Personas
- User journeys
- Sitemaps
- Wireframes
- Graphic designs
- User stories

Good training sites: http://theuxreview.co.uk/series/beginners-guide-to-ux/ http://www.uxapprentice.com/

UX Software

- Visual designs:
  - Pen and paper (free)
  - Balsamiq http://balsamiq.com/download/ (free for students)
  - Pencil http://pencil.evolus.vn/ (free)
- User stories:
  - Trello https://trello.com/ (free)

Personas

- Get to know the people who will use your system
- Understand their problem
- Understand how people already solve that problem
- Create **personas:** examples of each type of user
  - http://theuxreview.co.uk/personas-the-beginners-guide/

Ushahidi Persona

Guillermo // News Gatherer

"I work for a large news organisation, and we want to find new ways to source and tell stories. Crowdsourcing helps us get a better understanding of big events as they unfold. Publishing reports from citizens also helps us differentiate ourselves competitively."

Overview

Guillermo's job is focused on utilising social media for his news organisation. He uses social media to gather information about emerging events.

His goal is both help journalists source new and different stories, and also help connect the outlet better with its audience.

He uses Ushahidi on occasions when there is a big event, such as civil unrest or a natural disaster.

With this focus, he is prepared to invest time in getting to know Ushahidi. While he'd prefer everything to work perfectly right out of the box, he knows that it's important to customise things so it's more effective.

He's not a technical person, and so relies on the IT people at his office a lot to get the software up and running as he needs it. They can be slow sometimes, so he'd rather not depend on them.

Satisfiers

Getting a deployment up and running quickly.

Making sure the deployment is visually compelling and professional.

Making it easy for citizens to submit reports of all different media types.

Quick and accurate report verification.

Making it easy for journalists to uncover interesting and useful content.

Frustraters

Quality of reports is often low; poor descriptions or highly opinionated.

Journalists are often not interested in using Ushahidi to help source their stories; they sometimes don't see the value.

Usage scenarios

Configure deployment to have the right categories, verification schema, visual presentation.

Set up users with different editing permissions, and permissions to see different levels of information.

Define report structure and permissions.

Coordinate with verification and geolocation volunteer team managers to make sure the flow of reports are being processed.

Share sample outputs with management and journalists to help them start using the platform.

Periodically review the reports and outputs to make sure that everything is running correctly.

Technical literacy	●●○○○
Customisation needs	●●●○○

Deployment team	20-30
Reporters	500-1000

Report volume	100 per day
Deployment duration	2 months

smallsurfaces Ushahidi

User Journeys

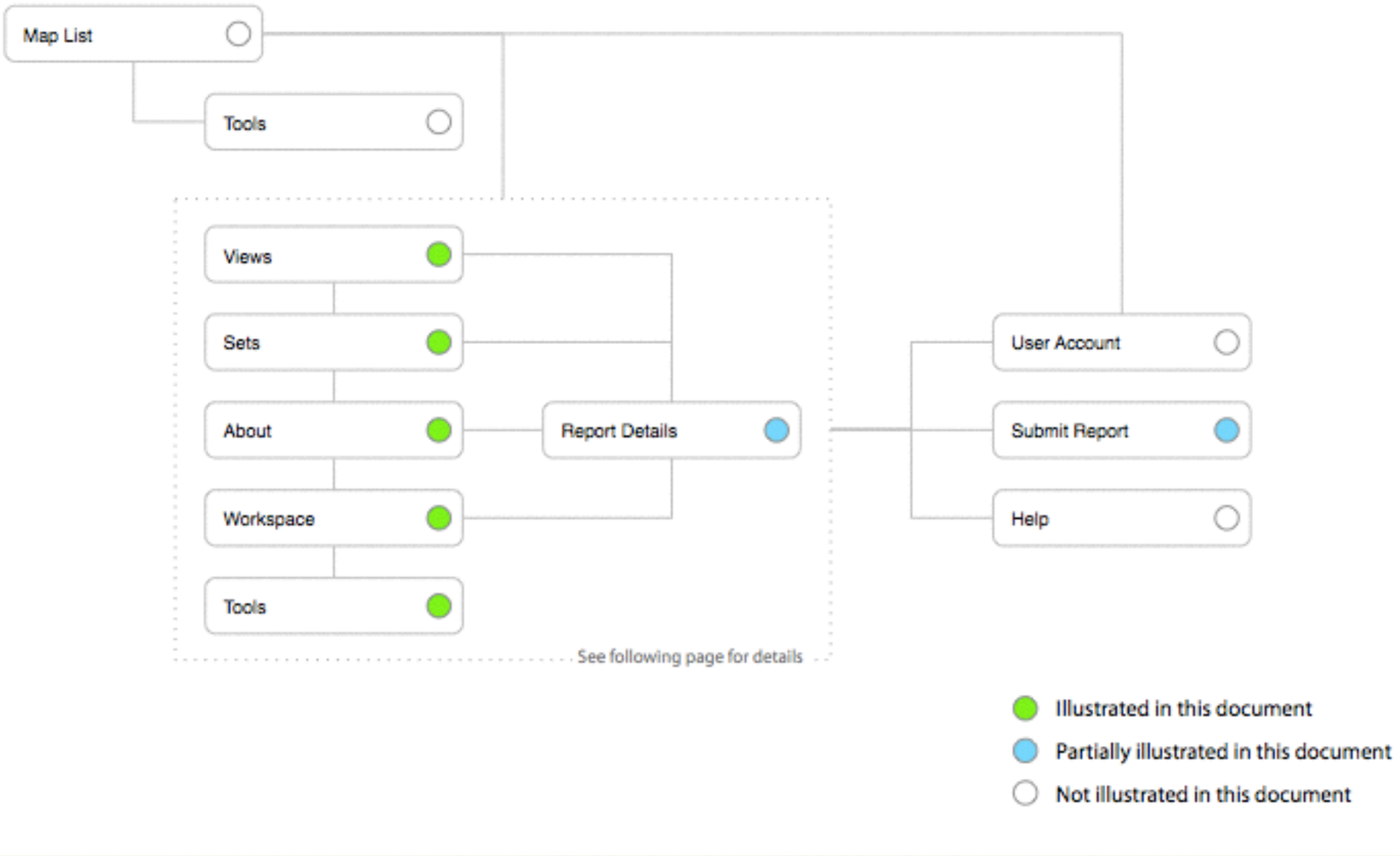
- Create **journeys (aka scenarios *)**: how the user will solve their problem
  - http://theuxreview.co.uk/user-journeys-beginners-guide/
- Contains:
  - Context - where?
  - Progression - what are the steps in this journey?
  - Devices - mobile? laptop? smartphone?
  - Functionality - what do they expect?
  - Emotion - angry? tired? scared? time-pressured?

* Not quite, but close enough: see http://ux.stackexchange.com/questions/30271/what-is-the-difference-between-scenario-and-user-journey

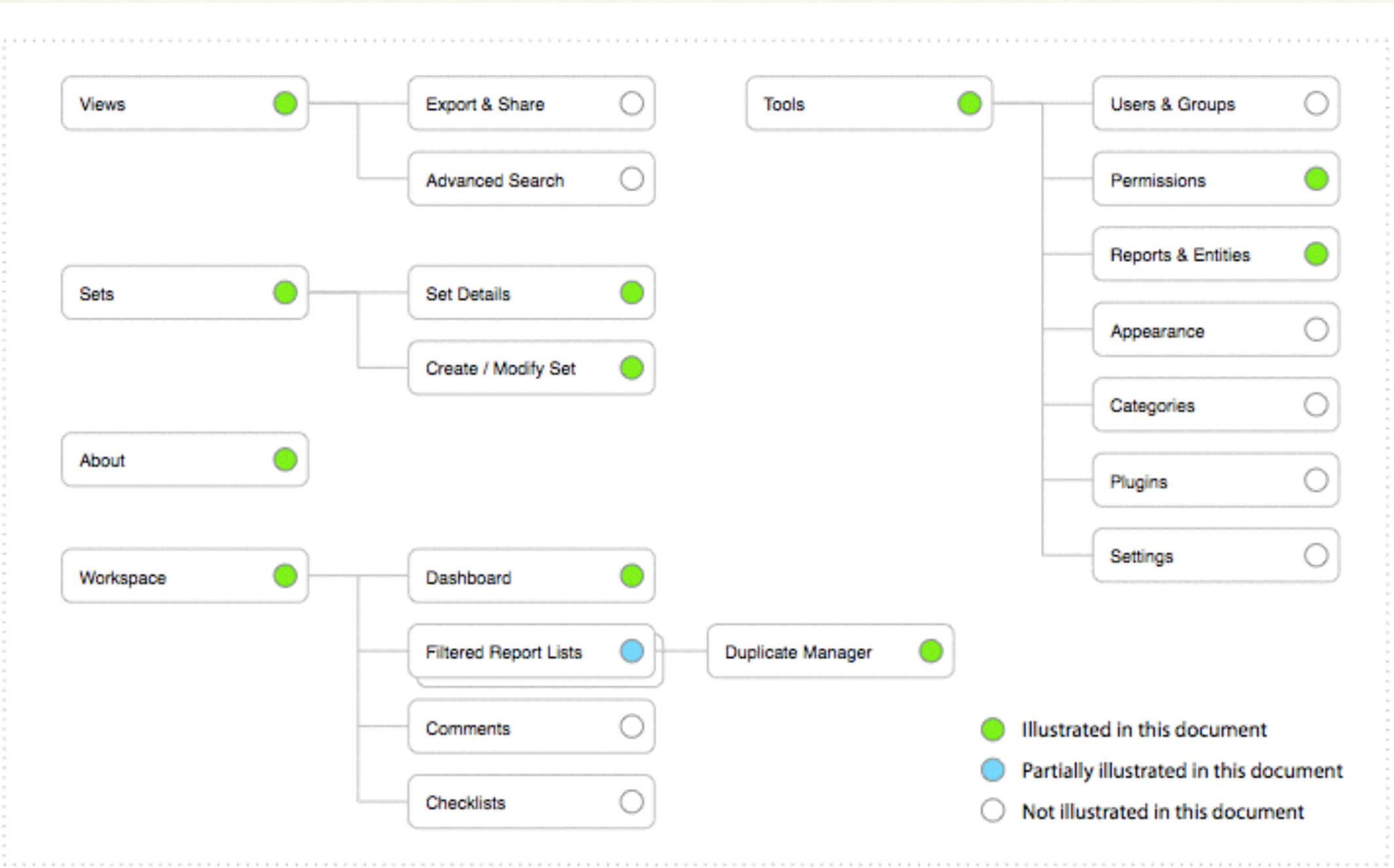
Sitemap

- What are all the pages on your site - how are they related?
- Can be generated by a “card sort”: write all your system functions (or user stories) on post-its, then sort them into groups

Ushahidi Sitemap



Ushahidi Sitemap pt 2

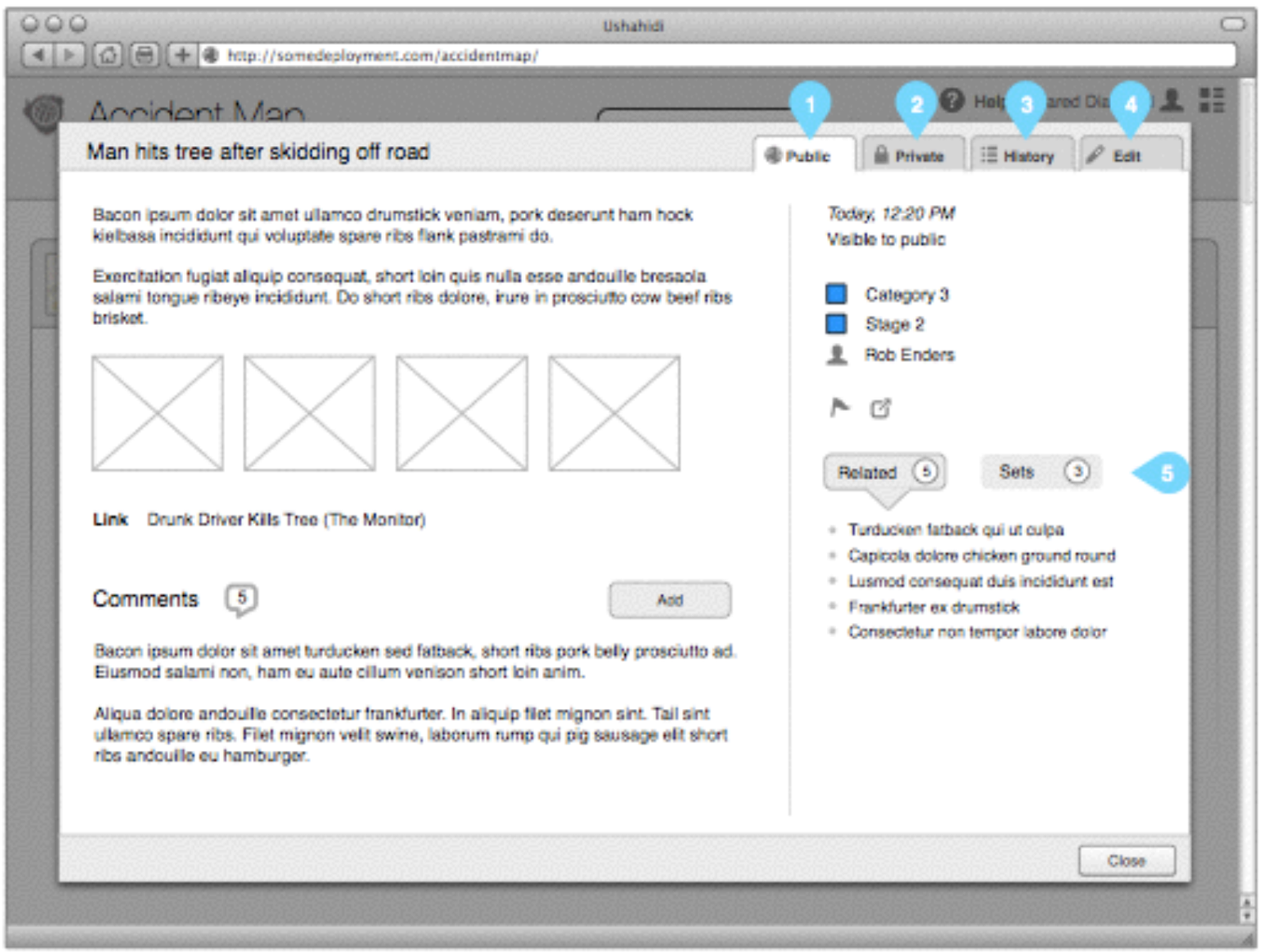


Page Design

- **Wireframe**: **Sketch** what's on each page.
- **Mockup**: closer to the final product, useful for walking through site functions with potential users
  - Clickable mockups are good
  - Can also use paper versions of the pages
  - NB. Balsamiq wireframes are clickable
- **Graphics**: look-and-feel of your site

Hint: go look at similar websites. See what you do and don't like about them

Ushahidi Wireframe



1. The Public tab shows how the report is represented to users who are not logged in. If reports are not visible to the public, this tab is not displayed.

   If the user is not logged in, none of the tabs shown here are displayed (i.e. only the Public view is shown).

   It is possible to reach this screen directly using a URL for the report (e.g. some deployment.com/accidentmap/r/43924

2. The Private tab shows all the report metadata (including both information published publicly, and information that is hidden from the public) in read only form.

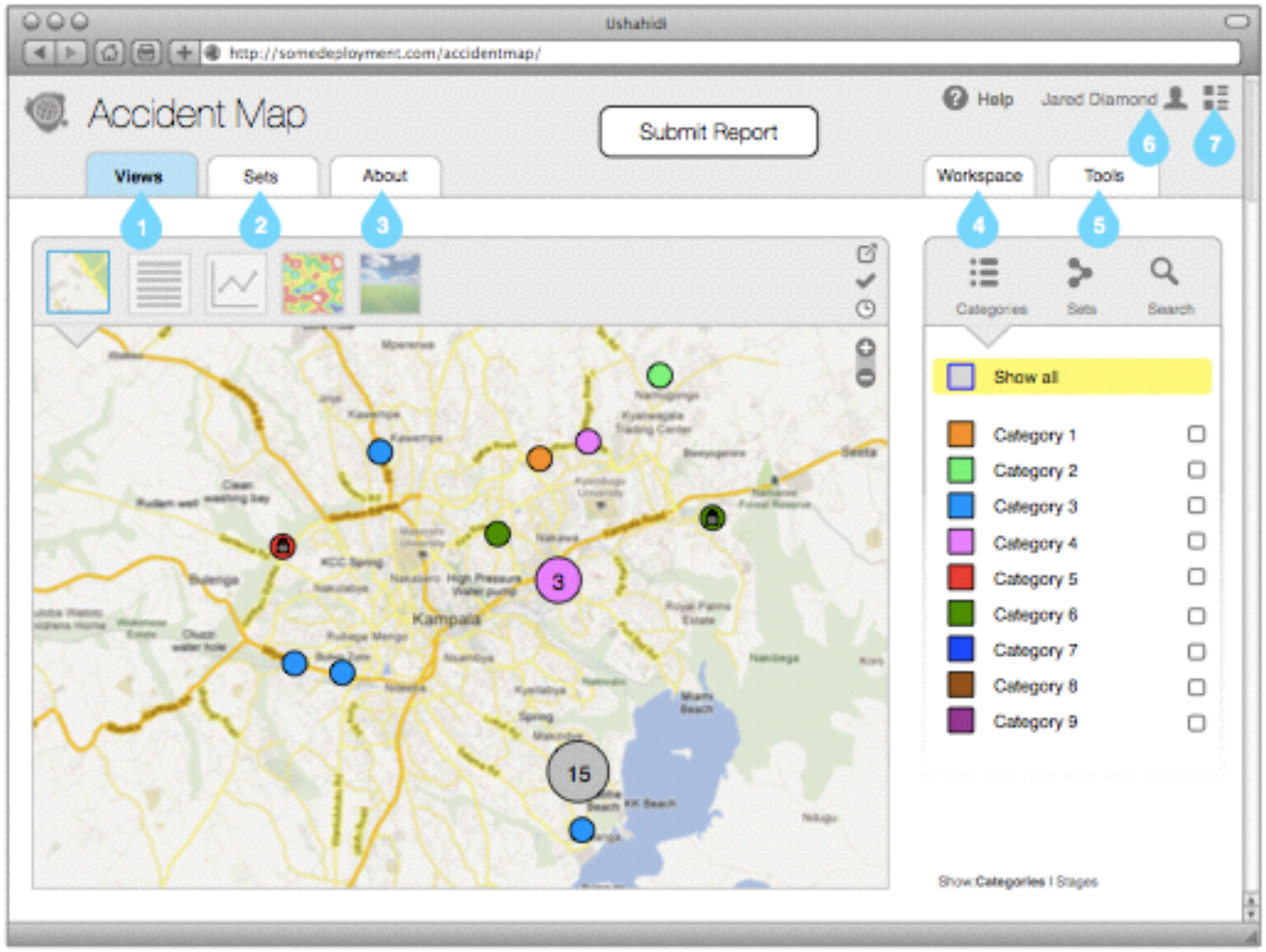
3. The History tab shows a reverse chronological history of changes to the report, including information about who made what changes.

4. The Edit tab allows users to modify any aspect of the report they have permission to change.

5. Users can explore other similar content, either reports that are related (generated computationally), or sets that this report is a member of.

Lorem Ipsum (random text): http://www.lipsum.com/

Ushahidi Mockup



1. The Views tab is where users can visualise, understand and interpret the data that lives in a deployment.

2. The Sets tab is where users can create unique groupings of subsets of the data based on search queries. For example, a set could contain reports that contain a particular keyword associated with a particular location.

3. The About tab is an example of a tab that has been created by a plugin. In this case, the plugin has created a tab that can be used for written articles or the display of specific media.

4. The Workspace tab is where registered users can manage all the content that moves through a deployment on a day-to-day basis. The Workspace is where users manage reports, comments and keep track of their progress against checklists.

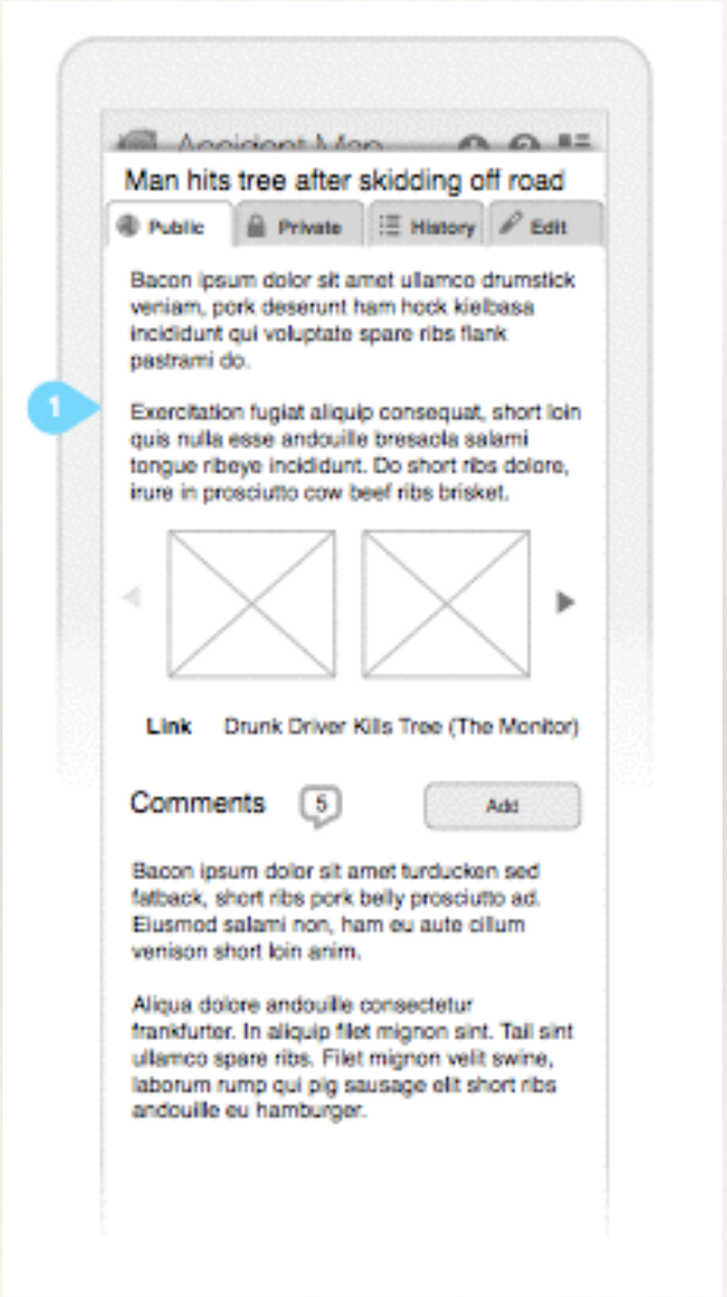
5. The Tools tab is where the system is configured and managed by those with administrative permissions.

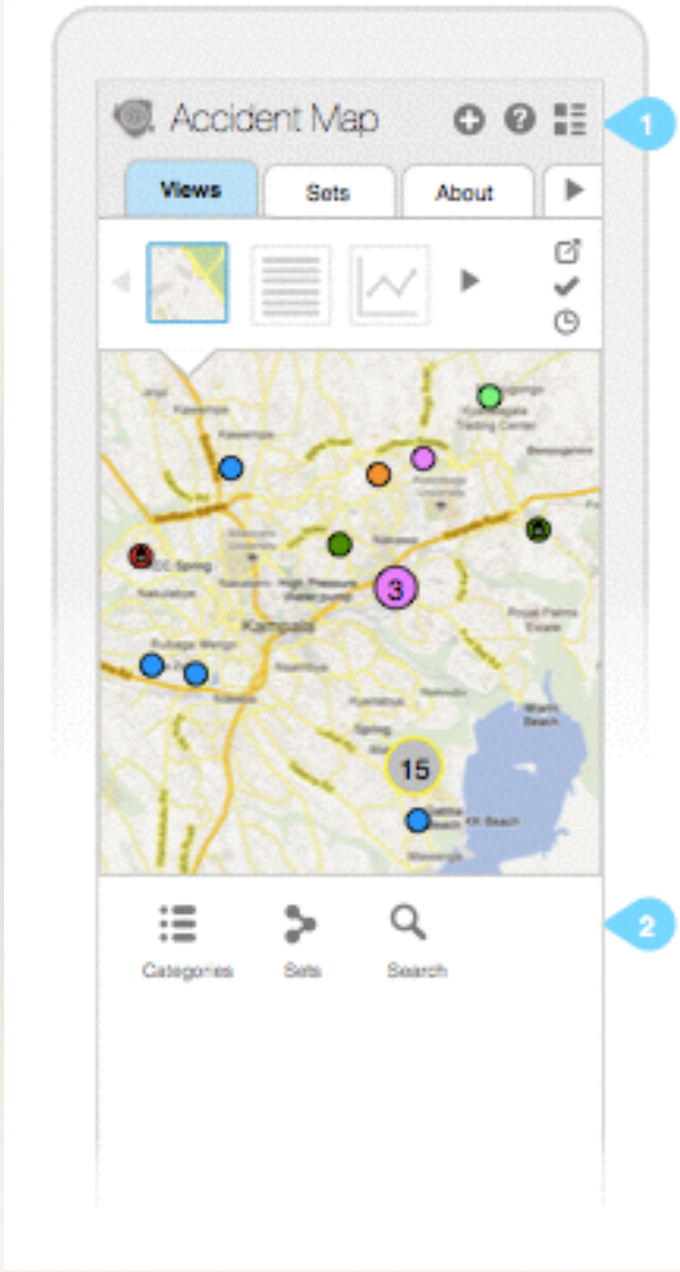
6. Logged in users have access to their profile (password and other account details) through the link of their name. When the user places the mouse over this link, a menu is shown allowing users to either view their account profile or log out.

   The user profile should allow users to specify which tab should be displayed by default upon login.

7. If the user clicks on the Map List icon, they are shown a list of all the maps available through this deployment. The user can also access account settings and overall deployment configuration through the Map List (not illustrated).

Don’t Forget Mobile…





Graphic Design

- Look-and-feel / Branding / Familiarity:
  - Color palette
  - Fonts
  - Shapes
  - Images (icons, header images, backgrounds etc)

- Either:
  - Know someone who can design and build all the graphic elements
  - Or use packages e,g, Bootstrap, Foundation with tools (e.g. DivShot)

Back-End Design

- So far, we've been designing the **front-end** of the system
  - e.g. the things that the user can see
  - and the interactions the user has with the system
- The **back-end** is the code that makes this all work
  - we'll be learning a lot of this today
  - NB: if it's complicated, there's usually a library for that.

User Stories

- Used for planning coding “sprints”
- Look like this:
  - **As a** <role>
  - **I want to** <goal>
  - **in order to** <benefit>

Agile Development

- Build fast, get feedback fast
- “Sprints” <- short cycles of coding and feedback
- People:
  - Product owner: defines features, sets requirements
  - Scrum master: gets system built
- Resources:
  - User stories
  - Kanban charts

Variant: BitsFeaturesTruth

Kanban

- Those “cards” on office walls…
- 5 columns: backlog, ready, coding, testing, done
- User stories and tasks in these columns
- Pick the n most important cards for each cycle (“sprint”)
- Move each card from left to right

Useful (free) software for this: Trello

Minimum Viable Product

- Build the *smallest* system that you can build and get feedback on
- Get feedback
- Adapt your design
- Repeat

Example: Data Repository

- Design brief:
  - I have a *lot* of datasets and pointers to datasets in my collections.
  - I'd like to make them available to more people, but they get lost in big data repositories and existing repository managers are too complicated and difficult to manage for this task.
  - I want people to be able to find datasets related to a given topic, add datasets of their own, and know where the **cleanest** copy of a dataset is online.
  - I also want people with low bandwidth to be able to access dataset lists easily.