Stakeholder Map

A network diagram of the people involved with or impacted by a proposed system design.

The stakeholder map helps the team visualize the relationships, hierarchies, and interactions between all the people who have an interest in the system to be built. The team works together to generate a map of all the individuals or groups who will potentially touch the software system – from purchasing decisions to use, installation, maintenance, support, or repurposing. Typically you will find that there are more people with a stake in the system than the obvious end-users.

As a visualization aide the stakeholder can help the team decide who should be involved in follow-up discussions and what stakeholders (individuals or groups) may require additional research or attention. Stakeholder maps are also a great way to summarize findings from research and should not necessarily be limited only to discovery. They also provide a quick snapshot of the overall context of the system – great for bringing new teammates up to speed or as an aide in architecture validation.

Benefits

- Identify more than just the usual stakeholders.
- Create a shared concept regarding ideas about stakeholders.
- Document and guide plans for user research.
- Help keep the team focused on people rather than technologies.

Quick Guide

- Draw simple icons to represent individual people
- Label people by specific role
- Don't represent categories of people as a single icon
- Draw a speech bubble to summarize thoughts and feelings
- Draw arrows to connect people
- Label lines to describe relationships between people

Resources

- "Identifying and Analyzing Stakeholders and Their Interests" -- http://ctb.ku.edu/en/tablecontents/chapter7_section8_main.aspx
- "The Theory Underlying Concept Maps and How to Construct and Use Them" – http://cmap.ihmc.us/Publications/ResearchPapers/TheoryCmaps/TheoryUnderlyingConceptMaps.htm
- "Stakeholder Analysis: Winning Support for your Projects" -- http://www.mindtools.com/pages/article/newPPM_07.htm

Acknowledgements

- Ariadna Font | http://ariadna.font.cat/ | @quicola

System Properties Web

A brainstorming kick and system visualization that focuses on system quality attributes.

The System Properties Web helps teams to elicit, categorize, refine, and prioritize stakeholder concerns that can later be turned into quality attribute scenarios describing desired system properties. This activity can be used alone or as a part of a larger "mini-quality attributes workshop." Typically group members brainstorm raw quality attribute scenarios and categorize them on the web. The properties web can be pre-populated with typical quality attributes, or populated as a team just prior to starting the activity.

Arranging quality attributes in a web creates a kick start for brainstorming and helps keep the discussion focused on quality attributes. The web itself can be used as a visualization aide in two ways. During the workshop you can quickly get a feeling for stakeholders' concerns based on where sticky notes cluster on the web. You can also use the web to create a system "signature" by creating a radar diagram on top of the web.

Benefits

- Provide a focal point for facilitating a mini-quality attributes workshop.
- Guide stakeholders in thinking about system properties rather than functions and features.
- Provide a visualization that shows how one system is different from another based on the high highly desirable properties.
- Help stakeholders prioritize quality attributes concerns and scenarios before refining them.

Quick Guide

- Draw or post the "properties web" so everyone can see it. The web can be created ahead of time if your team deals with the same types of systems often, or brainstormed as a group just-in-time.
- Brainstorm concerns and raw quality attribute scenarios as a group, one per sticky note.
- Categorize the concerns and raw quality attribute scenarios on the web.
- Prioritize using dot voting, both scenarios and overall qualities.
- Refine concerns and raw scenarios into more formal scenarios as time allows for the workshop.

Use a "quality attributes taxonomy questionnaire" instead of brainstorming during step 2 if one exists. This works well for systems that your team has some prior experience.

Resources

- SEI Quality Attributes Workshop -- http://www.sei.cmu.edu/architecture/tools/establish/qaw.cfm
- Microsoft Application Architecture Guide, Chapter 16: Quality Attributes -- http://msdn.microsoft.com/en-us/library/ee658094.aspx

Risk Storming

A collaborative and visual technique for identifying risks in a software system's architecture.

Risk Storming is a workshop proposed by Simon Brown that uses architecture sketches as a kick for brainstorming system risks. Placing the risks directly on architecture diagrams creates a visualization that facilitates team discussion and shows portions of the system that require further mitigation actions or exploration through experimentation.

Diagrams can be sketched collaboratively just-in-time, or printed from the project workbook. At least one diagram each depicting static, dynamic, and physical views should be prepared. Context diagrams also work well. This workshop works best with around 3-5 sketches max so as not to overwhelm the team. Variants of this workshop can be used to generate ideas or validate the architecture include "Rose-Bud-Thorn" and "Question-Fact-Idea" respectively.

Benefits

- Quickly identify risks in the proposed system architecture.
- Visualize high risk, unknown, or concerning parts of the system.
- Constrain risk identification to architectural concerns.

Quick Guide

- Sketch or print a handful of viewpoints of current software architecture.
- Include multiple views depicting a range of structures (dynamic, static, and physical) as well as context views.
- Brainstorm risks individually putting one risk per sticky note.
- Use different colored sticky notes to depict high/medium/low "exposure". This is somewhat subjective and based on probability, risk, and timeframe.
- Cluster sticky notes on the diagrams according to a risk's source, condition, or consequence.
- Discuss risks on the diagram as a team.
- Develop mitigation strategies and experiments for high priority risks.

Resources

- "Risk Storming: A Collaborative Visual Technique for Identifying Risk" -- http://www.codingthearchitecture.com/2012/07/11/risk_storming.html

Acknowledgements

- Simon Brown | http://www.codingthearchitecture.com | @simonbrown

Round Robin Design

An ideation technique that promotes group authorship, exploration, and builds consensus.

Round Robin Design is an activity borrowed from the User Experience community and adapted for software architecture design ideation. This workshop works well with a 2-5 small groups of 3-5 people but can also work designing solo. The goal of this workshop is to generate many different ideas. Placing constraints on the design process encourages the team to defer snap judgments while trading designs for critique and improvement helps to build consensus and a sense of group ownership.

After the initial system architecture designs have been critiqued and improved, divide the team into groups (if they aren't already) and summarize ideas by creating posters. Present the posters and have the team vote on best overall poster and "best design element" using multi-voting and visualize-the-vote techniques. This helps you to quickly identify important themes (best poster), but also potentially important design details that came out of the exercise.

Benefits

- Foster creativity by constraining the design environment.
- Create opportunities for unintended combinations.
- Encourage group ownership of the design.
- Build consensus among possibly disparate ideas.

Quick Guide

- Choose an architectural perspective to explore (e.g. static, dynamic, physical, context).
- Review relevant architectural drivers with the team.
- All participants sketch a solution. Unconventional ideas are encouraged.
- Pass your sketch to the left. Using a different colored pen, critique the sketch.
- Pass the sketch to the left. Using a different colored pen, improve the system design to overcome weaknesses identified by the critique.
- Review the sketches as a group and discuss.

Acknowledgements

- Ariadna Font | http://ariadna.font.cat/ | @quicola