A Concept Map for App Inventor

Sara Burns
Computer Science
Wellesley College
sburns@wellesley.edu

Project Supervisor: Franklyn Turbak
Associate Professor of Computer Science
Wellesley College
fturbak@wellesley.edu

Introduction

Computational Thinking Through Mobile Computing is an NSF-funded project that explores using App Inventor to teach key concepts in computer science in the context of developing mobile apps. Computational thinking [1] applies the type of thinking used in computer science, like iteration, recursion, abstraction, and automation, to conquer problems in every domain. As users create apps with App Inventor, they utilize many of these concepts, but their encounters do not give them a concrete knowledge of the topics. We believe that by explicitly introducing users to these concepts and connecting the reality of app creation to the theory and detail behind it, learning will be enhanced.

A concept map is a diagrammatic representation of knowledge that emphasizes the relationships between concepts. Typically, it consists of concepts as nodes in a graph, connected by edges that are labeled with linking words [2]. Often, the educational focus is on teaching students to construct concept maps in order to independently structure their knowledge of a subject. However, we believe that navigating this preexisting concept map will still aid user understanding.

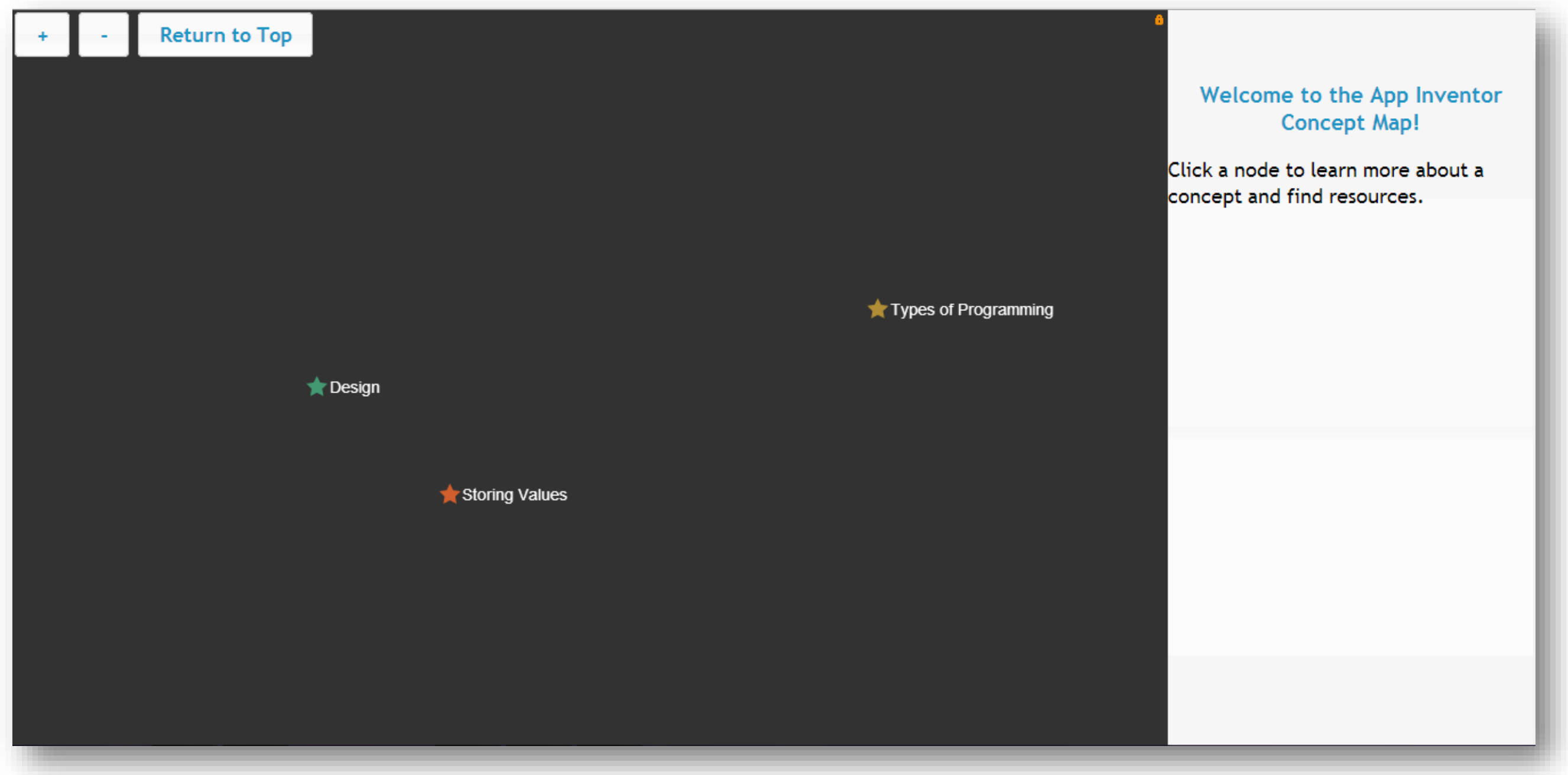
Project Goals

- Create a concept map of computing concepts encountered in App Inventor, highlighting the relationships of these concepts both to each other and to their role in app creation
- Develop useful resources for each concept, including a definition, links to helpful tutorials and examples, and related code "nuggets" (small blocks of App Inventor code that illustrate the concept)
- Create an interface that is easy and enjoyable for the user to navigate
- Create an interface for streamlined editing by *concept cartographers* (the people constructing the concept map)

Future Work

- Flesh out the current simple prototype of the map to include a wide range of concepts
- Develop an extensive collection of code nuggets for use in the map
- Make the map searchable
- Try out code nuggets in App Inventor with new template feature
- Record & analyze usage data

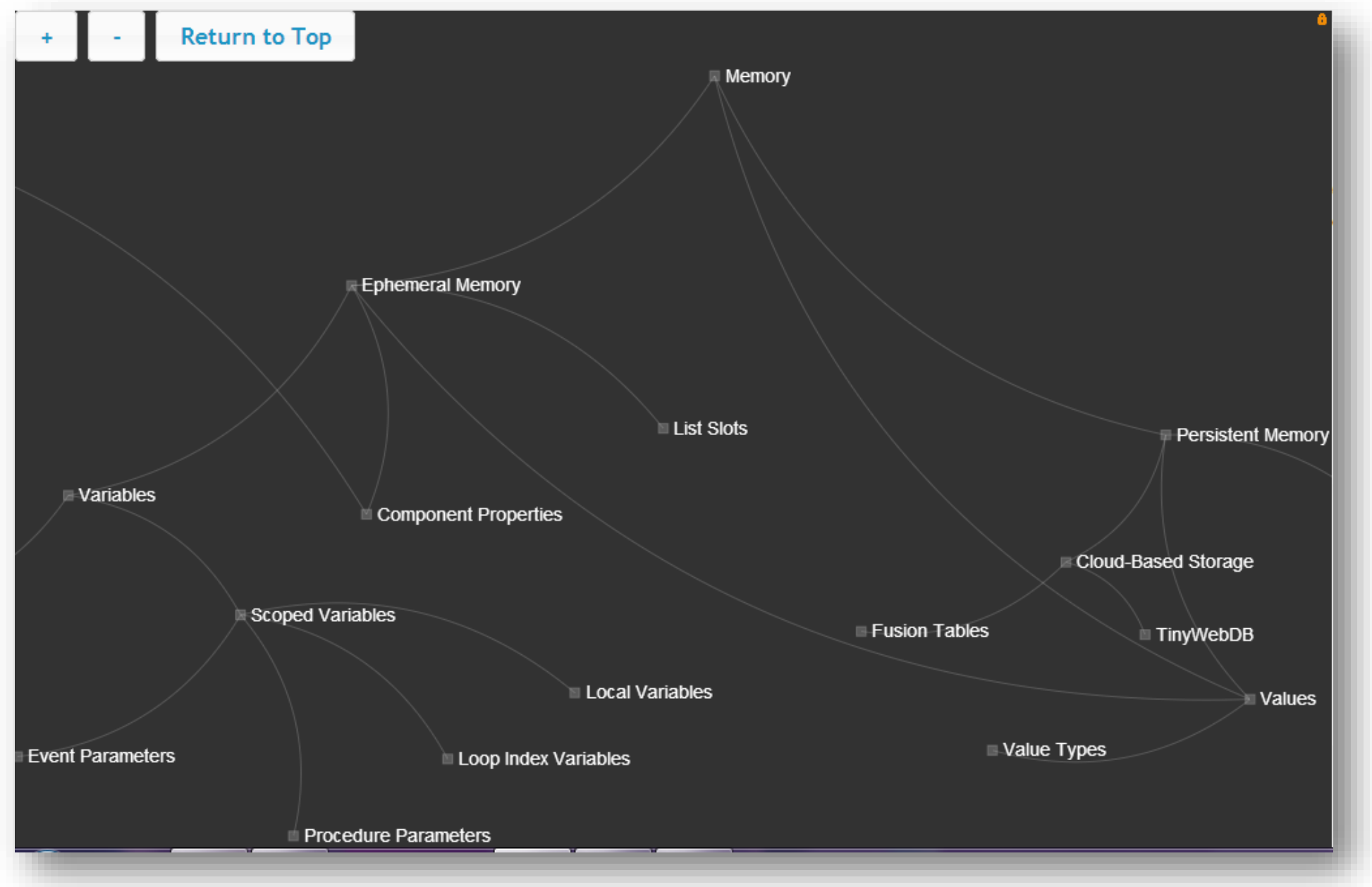
User Experience



Welcome to the App Inventor Concept Map!

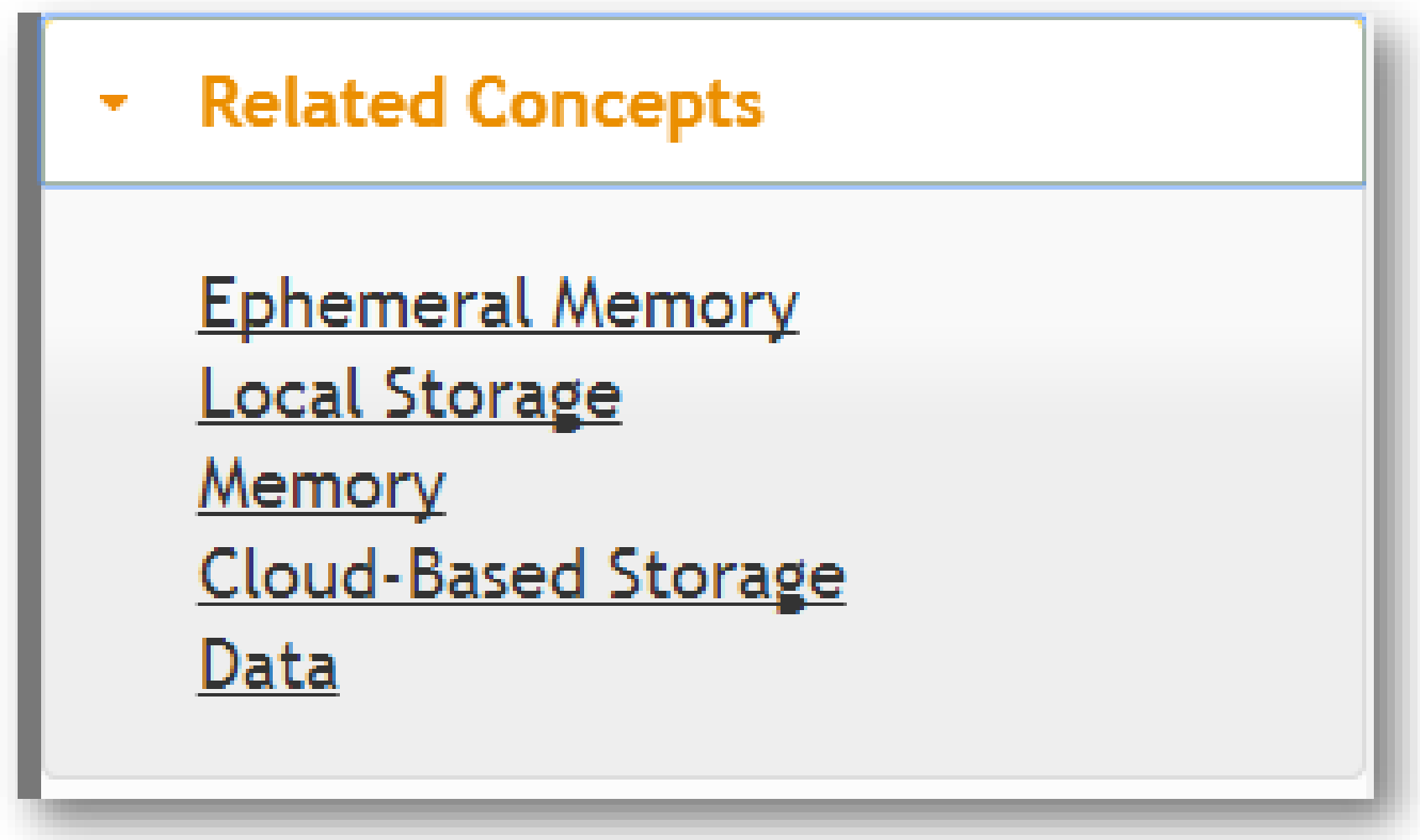
Click a node to learn more about a concept and find resources.

The landing page. Users are first presented with large ideas. When they click on an idea, they are taken to a network of related concepts.



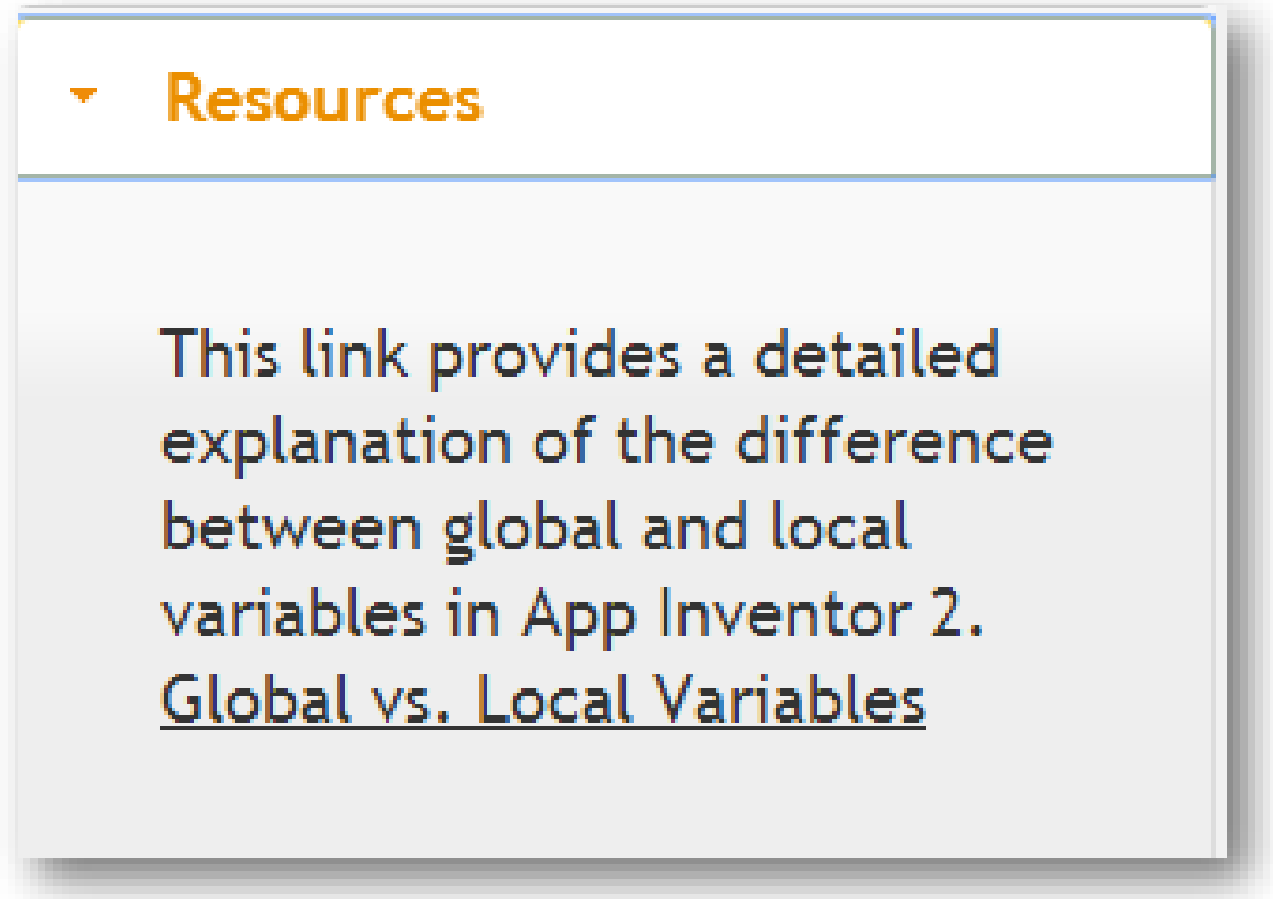
The content level of the map . The map is zoomable, and when a node is clicked on, adjacent nodes are highlighted and other nodes in the graph are faded. The information for the selected node is displayed in the information panel.

One way to navigate through the map is visually, clicking on different nodes. Another way is to use the links provided in the Related Concepts section.



Related Concepts

Ephemeral Memory
Local Storage
Memory
Cloud-Based Storage
Data

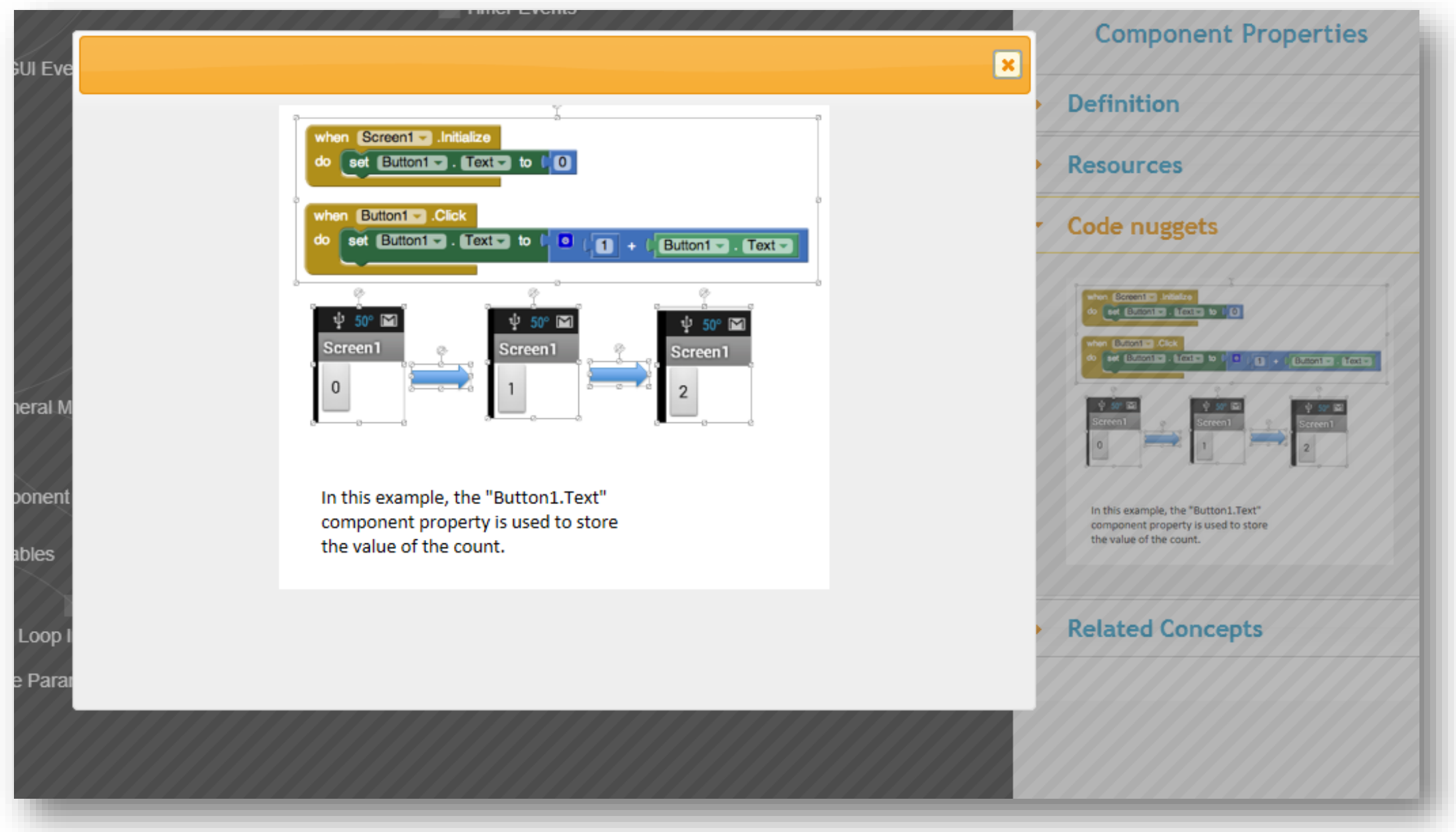


Resources

This link provides a detailed explanation of the difference between global and local variables in App Inventor 2. Global vs. Local Variables

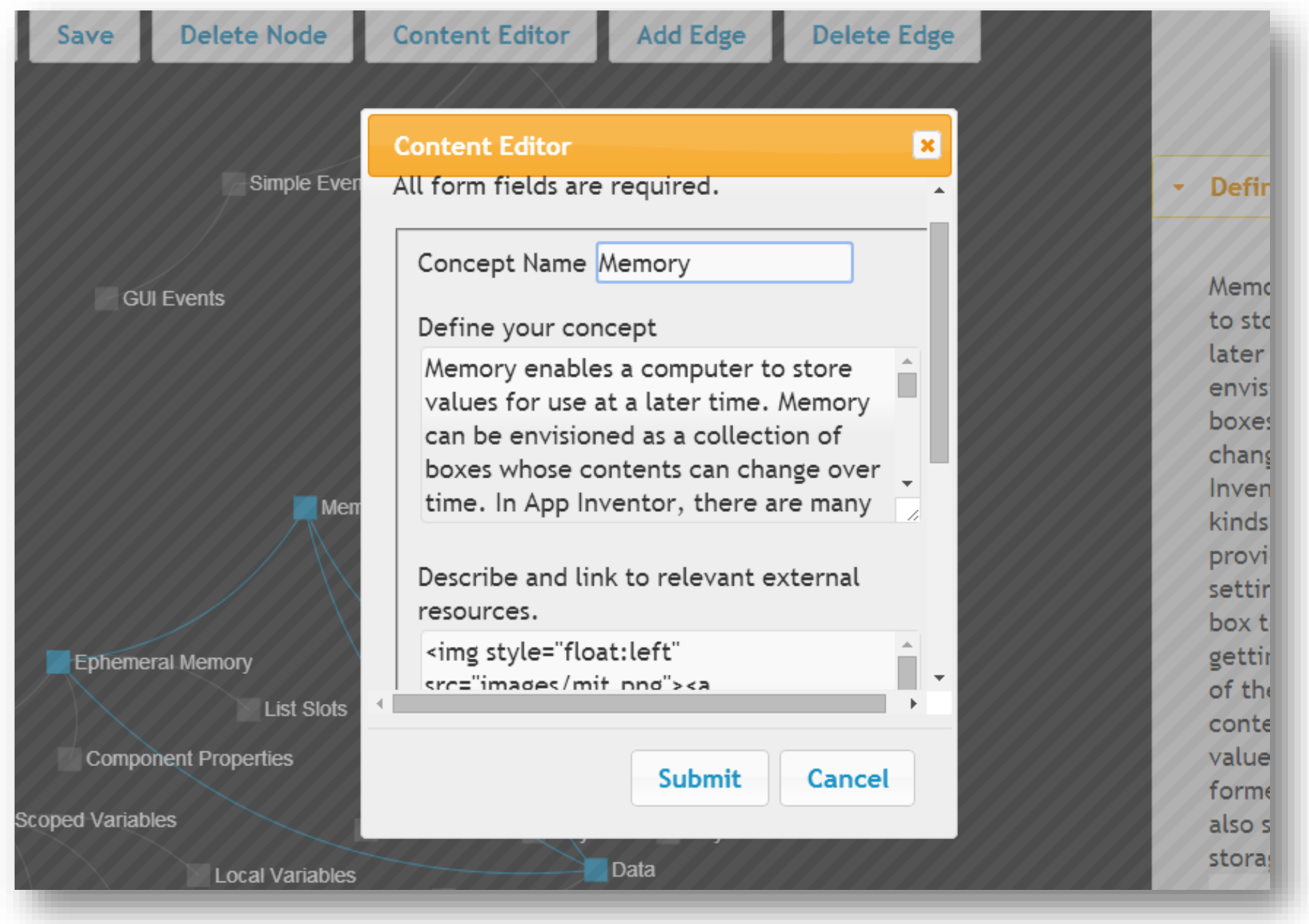
A close-up of the Resources section.

Code nuggets are displayed in miniature in the info window. When clicked, the nuggets are expanded.



In this example, the "Button1.Text" component property is used to store the value of the count.

Edit Mode



Content Editor

All form fields are required.

Concept Name: Memory

Define your concept: Memory enables a computer to store values for use at a later time. Memory can be envisioned as a collection of boxes whose contents can change over time. In App Inventor, there are many

Describe and link to relevant external resources.
<img style="float:left" src="images/mit.png"><a

Submit Cancel

Concept cartographers build the map using a special Edit Mode. Nodes can be created by double-clicking the canvas, and rearranged by dragging across the screen. Edges may be added and deleted by dragging the mouse from source to target. The Content Editor allows information to be entered and revised. Nodes may be deleted by clicking the delete node button or by hitting the delete key. The map is saved in the cloud each time content is edited, and can also be manually saved.

Implementation

The interface is written primarily in JavaScript. It utilizes the JQuery framework, JQuery UI for much of the design, and Sigma.js for the graphing features. The Sigma.js library was extended to create many of the editing features. The graph is saved to the server as a JSON file using a Python CGI script, and then loaded from the JSON file into the canvas using the graphing library. The HTML page is created dynamically using another Python CGI script, so that any HTML added while editing is included.

Acknowledgements & References

Computational Thinking Through Mobile Computing is a collaborative project among Wellesley College, MIT, Trinity College, University of Massachusetts Lowell, and University of San Francisco. This concept map work was funded by the Wellesley College Science Center Summer Research Program and National Science Foundation Grant DUE-1226216. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

[1]Wing, Jeannette M. "Computational thinking." *Communications of the ACM* 49.3 (2006): 33-35.
[2]Novak, Joseph D., and Alberto J. Canas. "The Theory Underlying Concept Maps and How to Construct and Use Them[1]."Institute for Human and Machine Cognition. N.p., 1 Jan. 2008. Web. 10 July 2014.