Village Volunteers Calendar  
October 19th, 2007

The source code is located at <http://code.google.com/p/vvcal/>. The current project owners are:

* Charlotte Tan
* James Staggs
* Holly Bowen
* Jeremy Stitz
* Andrew Slocum
* Josh Cronemeyer

These files and folders will be created when you checkout the project from SVN:

css - stylesheets  
docs - Excel file with user stories  
images - a one pixel gif necessary for Google authentication  
js - JavaScript files where most of the application logic resides  
jsunit - file needed to use jsunit for unit testing  
spikes - sample code created during development; not in use  
tests - a jsunit test script that demonstrates how unit testing has not worked for us in this scenario

dbinfo.php - information needed to connect to MySQL database  
index.html - handles Google auth, creates iframes for our component and the calendar  
thoughtworks.html - our component, creates iframes for the Villages and Volunteers tabs  
villages.php - markup for the Villages tab  
volunteers.php - markup and server-side script for the Volunteers tab

The auth.js file handles authentication with Google via AuthSub, so that a token is passed with each GData request. It is included in index.html, and the second line sets the init() function to run as a callback when the page loads. We found that this will conflict with any body onload events elsewhere in the application, so we opted for alternatives such as button onclick. Our component and the calendar are only displayed when the user is logged in with a token. Note that this login is separate from the regular Google account login.

The villages.js file starts when the ‘Go’ button is pressed in the Villages tab. The Google Calendar API uses continuation passing style, so when refreshVillageItinerary() calls getAllCalendarsFeed(), the second parameter is the name of a callback function that will be executed with the results. The SCOPE defined must be wide enough to include all of the Calendars the application will access. The FEED defined is the URL for the desired calendars, based on which Google account is currently logged in.

The volunteers.js file works similarly, however there is additional script to handle page refreshes to facilitate the server-side processing.

calendarDateInput.js and tabber.js were downloaded for date selections and tabbing functionality.