2012

Configuration Management Plan

Hatfield Hall Application

This document covers the overall management structure and hierarchies backing the design of the application for Rose-Hulman's Hatfield Hall.

Ali Almajed, Caleb Drake, Andy Milluzzi
RHIT Software Quality Assurance
2012 April 13

Table of Contents

1 Software Configuration Management Plan

1.1 Reference Documents

All project files relating to the Hatfield Hall App (HHA) project can be accessed via the GitHub repository [1].
Documentation log can be found via Google Docs [2].

1.2 Management

1.2.1 Organization and Responsibilities

The HHA team shall equally share the responsibility of monitoring the software configuration.

Each feature shall have a team member assigned as *in-charge* of its overall implementation; however, other HHA members may help as needed by the member *in-charge*.

Sriram Mohan, the project advisor, may make recommendations to these management policies; however, the HHA team is ultimately accountable for implementing any alterations.

The HHA team maintains liability for any software-related consequences resulting from the direct use of the HHA under the Android 2.1 API [3].

1.2.2 Policies, Directives, and Procedures
○ Minutes must be uploaded to GitHub[1] within 24 hours after team/advisor/client meetings
○ Repository must be available to all team members, the advisor, and--if requested--to the client
○ The HHA must adhere to the code conventions set forth by Oracle Corporation[4]

1.3 Activities

1.3.1 Configuration Identification

1.3.1.1 Identifying Configuration Items
○ Meetings minutes: all meeting minutes will be documented in a Google-document. The meetings that will be documented include team meetings, advisor meetings and client meetings.
○ Project related files and the source codes will be stored mainly on the GitHub repository [1].
○ Potential designs and rough drafts will be stored on the GitHub repository [1].
○ A schedule of tasks and deadlines will be documented in the main Google-document [2].
○ Individual tasks will be documented in the main Google-document [2].

1.3.1.2 Naming Conventions

All of the documents that are stored in the GitHub repository [1] or Google Docs will follow a defined naming and shelving guidelines as follows.
- Project source codes and programming-related files should be limited to the following folders, HatfieldHallApp/ gen, HatfieldHallApp/ res, HatfieldHallApp/src, and HatfieldHallApp/ AndroidManifest.xml.
- Designs will each have a separate folder inside HatfieldHallApp/ Documents/Designs.
- CamelCase should be used in naming all the files followed by the milestone number (e.g. TestCases2.docx).
- Source codes and programming-related files will follow the Code Conventions for the Java TM Programming Language [4].

1.3.1.3 Acquiring Configuration Items

All of the files will follow the shelving and naming conventions stated in the previous heading except for the following exceptional cases:
- Multiple instances of a related thing, home-screen designs for example, should have the same name followed by a counter (i.e. HomeScreen1.jpg, HomeScreen2.jpg).
- There will be a Temporary-files folder HatfieldHallApp/Documents/Temp that will hold unfinished or temporary files.

1.3.2 Configuration Control

1.3.2.1 Requesting Changes

Due to the scope of the project, non-trivial changes (such as adding a feature), besides bugs, will require the discussion of the entire team and will be added to the main project Google-document's [2] feature list.

The format of changes will follow the current format of the document and include a summary, due date, and owner.

1.3.2.2 Evaluating Requested Changes

The main project Google document [2] will also contain a bugs list.

Upon discovery of a bug, it will be added to the list. The list will include a summary of the bug, severity, owner, and current status.

A new feature will be discussed with the group as described in the previous section. Upon approval of the change, it will be added to the current features list.

1.3.2.3 Approving or Disapproving Changes

All bugs are assumed to be accepted unless one group member feels need for discussion. Should the latter be the case, the status field—within the Google-doc [2] will be changed to "pending discussion".

Bugs pending discussion or new features will require a ⅔ majority to be implemented.

A bug that is not going to be fixed will be changed to the status of "known issue" and may be re-evaluated at a later time.

1.3.2.4 Implementing Changes

Once a change is made, the branch is to be merged to the master and the commit comment will reflect—with a direct copy from the Google-document [2]—the summary of the change made. The status field will be updated to "fixed" for a bug or "implemented" for a new feature.

1.3.3 Configuration Audits and Reviews

Configuration audits will be conducted at random—but at least once this term. Any team member may perform a full audit. Team members may also perform more frequent informal reviews.

1.3.4 Subcontractor and Vendor Control

The software is based on Android 2.1 API [3]. No future versions of the API are being used. Thus, there is no external code to control.

1.4 Resources

1.4.1 Schedules

Schedule information shall be expressed using absolute dates, dates relative to other project activities, project milestones, or a simple sequence of events.

All schedules are maintained within the main Google-document [2].

1.4.2 Resources

The tool used for the configuration management shall be GitHub [1].
Application development shall occur within the Eclipse Development Platform [5].
The project's primary data source is the Hatfield Hall Website [6], which uses HTML5 [7].

1.5 Plan Maintenance

The monitoring and management of the Software Configuration Management Plan shall be the responsibility of Caleb Drake.
- ○ Maintenance shall occur after bimonthly
- ○ Maintenance shall encompass--but not be limited to--the following
  - i. updating the Management Plan to reflect approved changes
  - ii. ensuring adherence of documents to the Management Plan
  - iii. correcting / delegating correction of documents not reflecting standards
- ○ All alterations must be proposed for discussion and approval at the first meeting following the aforementioned changes for which each HHA member is present
- ○ All approved changes shall be recorded within the minutes of the latter-mentioned meeting

2 References

1. Hatfield Hall App GitHub Repository:
   <git@github.com:08milluz/HatfieldHallApp.git>.
2. Real-time Hatfield Hall App Documentation:
   <https://docs.google.com/document/d/1vJspwmwVR1EXHExt128xXyuz6azsfiQXrE45e7TZh34/edit>
3. Android SDK:
   <http://developer.android.com/>.
4. Oracle Code Conventions for the Java TM Programming Language:
   <http://www.oracle.com/technetwork/java/javase/documentation/codeconvtoc-136057.html>.
5. Eclipse Platform:
   <http://www.eclipse.org/downloads/packages/eclipse-classic-372/indigosr2>
6. Hatfield Hall Website:
   <http://hatfieldhall.com/>.
7. WHATWG Community [HTML5 Assistance]:
   <http://developers.whatwg.org/>.