Configuration Management Plan

What's HappNin

Lauren Finley, David Lucero, Sumaiya Rashid,
Yavuz Kocoglu,Rehan Ayoub, Andres Suarez, Zack Luckman

Bashes

May 6, 2018

Table of Contents

Table of Contents **1**

1. Introduction **2**
1.1 Purpose 2
1.2 Scope 2
1.3 Definitions, Acronyms, and Abbreviations 3
1.4 References 4
1.5 Document Conventions 4

2. SCM Management **4**
2.1 Organization 4
2.2 Responsibilities 4
2.3 Applicable Policies, Directives, and Procedures 5

3. SCM Activities **5**
3.1 Configuration Identification 5
3.2 Configuration Control 6
3.3 Configuration Status Accounting 7
3.4 Configuration Evaluation and Reviews 8
3.5 Interface Control 9
3.6 Subcontractor/Vendor Control 9
3.7 Release Management and Delivery 9

4. SCM Schedule **9**

5. SCM Resources **10**

6. SCM Plan Maintenance **10**

1. Introduction

1.1 Purpose

The purpose of this Configuration Management Plan document is to identify and elaborate in detail the Configuration Management activities that are to be used during the system life cycle of the What's HappNin system.These activities include but are not limited to planning, implementing, submitting, and approving changes to the What's HappNin system. This document will be updated as necessary when the need arises.

1.2 Scope

The What's HappNin system consists of a smartphone application that will run on Android devices. This Configuration Management Plan applies to all components that are related to or interact with the What's HappNin system. It covers all changes made to the What's HappNin application- to include all services, hardware, software, and processes within the What's HappNin system.

1.2.1 Project Overview

The purpose of the What's HappNin (WHN) application is to allow users to connect with their friends by sharing the activities that they are doing. The users can achieve this by utilizing a map which pinpoints the locations where all of the user's friends are gathered and partaking in a social activity. When the user opens the app, they have the option to create an activity themselves by making a series of posts which indicate what kind of social activity the user is partaking in. Furthermore, the user will have the ability to upload comments, pictures or videos to keep their friends updated on the status of the event and entice friends to join. This will be accomplished on a map by including small features such as event icons and notifications that notify other users when posts are being made and of the location in which the activities are occurring. More specifically, the event icons on the map will include a symbol such as a basketball to let their friends know what type of event is taking place, the number of people of attending the event, and if clicked the event icon will redirect the user to the event feed. From this feed WHN users can see photos, comments, and videos that have been posted to an event. There will also be a point system in place to award users for creating or attending social events with their friends.

1.2.2 Additional Software

The What's HappNin application will integrate with the following external software components:

a. Google Maps
b. Android Mobile Operating System

1.2.3 Additional Hardware

There are no additional hardware components other than the smartphone that is required.

1.2.4 Limitations and Constraints

Time: The software development cycle of the What's HappNin application is limited to 3 months. This includes planning, documentation, implementation, and testing.

Resources: Software development tools are limited to free and open source tools due to lack of funding.

1.2.5 Assumptions

The following assumptions may impact the cost, schedule, or ability to perform outlined SCM activities:

a. The team of individuals overseeing the What's HappNin system consists entirely of college students. The degree of their participation in SCM activities varies depending on their individual class/work schedules.

1.3 Definitions, Acronyms, and Abbreviations

1.4.1. Definitions

Baseline: Specifies the components versions, libraries, configuration files that are used in a specific system.
Configuration: The functional characteristics of software as set forth in technical documentation.
Branching: When a source code worked on independently by different develops, and creating a different version of it.
Codeline: Different versions of source code or components of systems derived from earlier versions.
Configuration control: Process that ensures the all of the previous versions of systems and components are recorded and maintained.
Configuration item/software configuration item: Software project related stuffs that have been placed under configuration control. Configuration items have different versions and unique names.
Mainline: A sequence of baselines representing different versions of a system.
Merging: The process of combining different versions of separate codelines to create a new version of software component.
Release: A version of a system that has been delivered for usage to customers or other users.
Repository: A shared database of versions of software components and information about changes to these components.
System building: The creation of an executable version of system.
Version: A configuration item that differs from its previous and future types. Versions always have their own unique identifier.

Workspace: A private work area, which allows a developer to modify a software without affecting other developer's who are also modifying the same software.

1.4.2. Abbreviations

SCM: Software Configuration Management
CI: Configuration Item
SWCI: Software Configuration Item
CCB: Configuration Control Board

1.4 References

IEEE 828-2005 Std: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6044675

1.5 Document Conventions

This document was written using the IEEE 828 - 2005 Standard for Software Configuration Management Plans.

This SCM Plan conforms with the requirements of IEEE Std 828-2005.

2. SCM Management

2.1 Organization

The organizational unit for this project is the development team (Bashes). The development team is responsible for all of the SCM activities for the project, the problem resolution processes, as well as developing the whole system.

2.2 Responsibilities

The organizational unit Bashes shall be responsible for all SCM activities of the What's HappNin system.

Bashes
Purpose and objectives: The aims is to create an android app that allow the users to create or join a social event.

Membership and affiliations:

- The members of development team are Yavuz Can Kocoglu, Lauren Finley, Zack Luckman, David Lucero, Rehan Aoyub, Sumaiya Rashid and Andres Suarez

Period of effectivity: Each member of the development team were active from the beginning of the project and they are still active at the present.

Scope of authority: All members of the development team have equal authority over the project.

Operational Procedures:

Step 1: Plan the layout for the system.
Step 2: Decide on what baseline will be necessary to develop the system.
Step 3: Create the first version of a code, so the development team can divide the work among each other and they can develop the code on their own workspace.
Step 4: Keep the changes that were made to the codeline while building the system in a repository for future references.
Step 5: Build a final version of the system for users.

2.3 Applicable Policies, Directives, and Procedures

What's HappeNin system is subject to the external constraints of the Google Maps API. A detailed description of those constraints can be found the link below:
https://developers.google.com/maps/terms

3. SCM Activities

3.1 Configuration Identification

3.1.1 Identification of Configuration Items

The following Configuration Items will be managed by the What's HappNin system:

- D001v1 Software Requirements Specification document
- D002v1 Platform document
- D003v1 User Story document
- D004v1 Sprint Review document
- D005v1 Sprint Backlog document
- D006v1 Product Backlog document
- D007v1 Project Description document
- D008v1 Use Case Model document
- D009v1 Team Member Report document
- D010v1 Domain Model and Detailed Design document
- D011v1 GRL and UCM Models document
- D012v1 Zenhub Kanban Board

Note: We are still in the early stages of implementation. Additional items such as code files and test cases will be identified as we progress.

In order to track changes made to What's HappNin, we have established a configuration baseline called Baseline D. It has been determined that all system documentation will be placed under this baseline.

3.1.2 Naming Configuration Items

The naming of configuration items will be a unique combination of letters and numbers generated based on the baseline under which the configuration item is assigned to, the version of each configuration item, and the title of the configuration item.

For example the configuration item name for item D001v1 Platform document would be broken down as follows:

- The leading letter 'D' indicates that this item is under baseline D
- '001' is a unique identifier assigned to the configuration item
- 'v1' indicates that this is version 1 of the configuration item
- 'Platform document' indicates that the title of this configuration item is "Platform document"

3.1.2 Acquiring Configuration Items

The What's HappNin system will control the following libraries:

- Working Library
  - Supports code and development
  - All items from baseline D associated with the coding and documentation of the project will be controlled by this library
- Support Library
  - Supports testing
  - All test cases and test data will be controlled by this library
- Master Library
  - Supports finished product
  - Upon completion of the application, What's HappNin configuration items will be controlled by this library

3.2 Configuration Control

In this section the process of requesting, evaluating, approving/disproving, and implementing changes to configuration items for the What's HappNin system is outlined in detail.

3.2.1 Requesting Changes

To request a change to a baselined configuration item, the following information must be included as part of the request:

a. The name(s) and version(s) of the CIs where change is desired
b. Originator's name and organization

c. Date of request
d. Level of urgency
e. The need for the change
f. Description of the change being requested

Upon submission, each change request will be assigned a change request number and status to track each request.

3.2.2 Evaluating Changes

Each change request will be evaluated and reviewed on an individual bases. Changes will be evaluated based on the following criteria:

- Impact on project schedule
- Impact on project resources
- Benefits of implementing the change
- Consequences of not implementing the change

Each change request will be reviewed by the organization managing the system and evaluated based on their overall effect on the deliverable and resources.

3.2.3 Approving or Disapproving Changes

The configuration control boards that will approve proposed changes along with their level of authority are listed below:

- Bashes
  - *Members:* Yavuz Kocoglu, Lauren Finley, David Lucero, Rehan Ayoub, Zack Luckman, Sumaiya Rashid, Andres Suarez
  - *Authority:* Complete authority to approve or disprove all change requests. All decisions are final.
  - *Responsibilities:* Bashes is responsible for evaluating all aspect of a change request based on the criteria outlined in 3.2.2

3.2.4 Implementing Changes

Approved change requests must be verified and implemented by the development team. Change requests that have been approved will be documented with the following information:

- The request number of the approved request(s)
- The names and versions of the affected items
- Verification date and responsible party
- Release date or installation date and responsible party
- The identifier of the new version

3.3 Configuration Status Accounting

3.3.1 Data Elements and SCM Metrics

The following data elements and SCM metrics will be tracked and supported for baselines and changes:

- Name of Individual
- Date and Time
- Time of Change Request
- Description of change
- Associated baseline
- Lines of code
- Cost

3.3.1 Reports

The following reports are to be produced:

- Management Reports
  - Audience: Product owner or individual in management roles
  - Information: Approved changes
  - Frequency: At the end of each sprint
- QA Reports
  - Audience Needed: QA team
  - Information Needed: Test results, changes to test cases
  - Frequency: TBD
- CBB Reports
  - Audience: Bashes
  - Information Needed: Information pertaining to change requests
  - Frequency: At the end of each month

3.3.1 Information Collection

Information will be collected via a third party application. This application has not been decided yet at this point in time.

All reports will be stored via Google Drive.

3.3.1 Access to Status

Status data will be password protected. Each report will have a password associated with it that will only be known by the receiving party. You cannot view the reports without the correct password.

3.4 Configuration Evaluation and Reviews

Prior to its release, a configuration audit shall be performed on each CI. These audits will evaluate how well each CI meets the required physical and functional characteristics. These evaluations can be used by management to evaluate a baseline.

The audit shall define the following:

a. Its objective
b. The CIs under audit or review
c. The procedures for conducting the audit or review

d. The procedures for conducting the audit or review
e. The participants by title
f. Documentation required to be available for review or to support the audit or review
g. The procedure for recording any deficiencies and reporting corrective actions
h. The approval criteria and the specific action(s) to occur upon approval

3.5 Interface Control

What's HappNin does not contain any CIs that interface to items outside the scope of the plan.

3.6 Subcontractor/Vendor Control

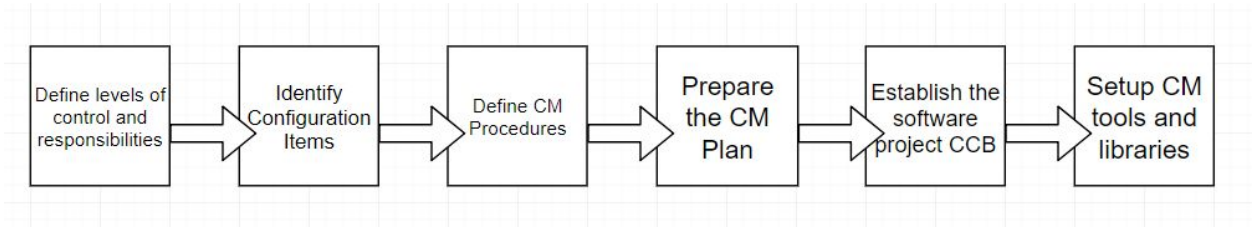
What's HappNin does not have any subcontractor/vendor control activities.

3.7 Release Management and Delivery

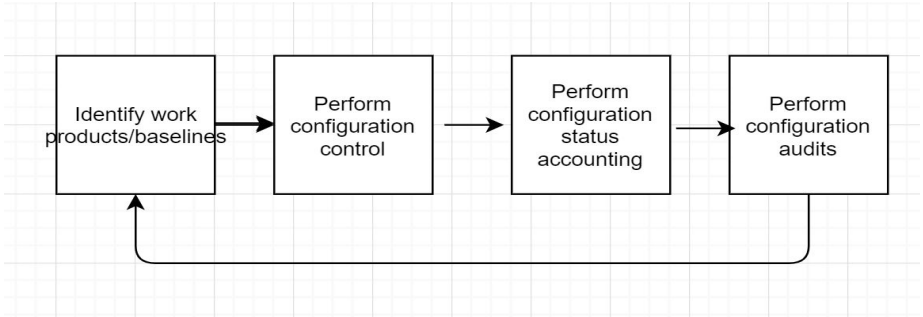
The build, release, and delivery of software products and documentation for the What's HappNin system will be formally controlled using GitHub. Master copies of code will be maintained for the life of What's HappNin. Sensitive or critical information will be encrypted before delivery to comply with the policies of involved organizations.

4. SCM Schedule

CM Planning Tasks



CM Execution Cycle



The SCM activities for the What's HappNin system will occur as a sequence of events as modeled above. The first phase will consist of planning, and during this phase tasks such as identification of CIs, preparation of the CMP, and establishment of software project CCBs occur. After the completion of the planning phase, the execution phase will begin. This phase is a cycle and will occur as many times as needed. This schedule does not include updates to the CMP that will occur on an as-needed basis.

5. SCM Resources

The software tools that will be used for this system are Google Maps API, Android Studio, Firebase, Google Drive, Github and Zenhub.

Google Maps API: This software will be used to create a real time map for the user to see.

Android Studio: A development environment for Google's android operating system. Android Studio will be used to write the code for the system.

Firebase: A database management system that is similar to MySQL. It will also handle the security aspect of the data.

Google Drive: A file storage application that allows the members of the development team share documents with each other.

Github: An application that allows the developers to share the code; also supports version control.

Zenhub: Allows the developer to add a Kanban board to the Github repository.

6. SCM Plan Maintenance

Lauren Finley is responsible for monitoring the plan and making sure that the rest of the development team follows the plan. The SCM plan will be reviewed at the end of each sprint, and updates to the plan are to be performed on an as-needed basis. If the need arises to make changes to the SCM plan, it will be discussed among the development team. After discussion, the decision as to whether or not the SCM plan will be changed will be voted upon by the development team.

History of Changes

Date	**Change**
3/9/2018	SCM Plan version 1 created

4/13/2018	Update
5/6/2018	Update