**Configuration Management**

for

trendAssist App

Version 1.1 approved

**Prepared by:**

Andre Manz

Cade Wall

Aaron Turner

Aaron Riggs

Xujia Wu

Mayur Bhakta

Flying Mongeese

March 2, 2018

**Index Page**

1. **Introduction……………………………………………………………... 3**
2. **SCM Management.……………………………………………………… 3**
   1. **Organization**
   2. **Activities and responsibilities**
   3. **Decisions process and responsibilities**
3. **SCM Activities………………………………………………………….. 4**
   1. **Configuration identification**
   2. **Configuration control**
   3. **Configuration status accounting**
   4. **Configuration evaluation and review**
   5. **Interface control**
   6. **vendor control**
   7. **Release Management and Delivery**
4. **SCM Schedule…………………………………………………………… 6**
   1. **Sequence and coordination of SCM activities**
5. **SCM Resources………………………………………………………….. 7**
6. **SCM Plan Maintenance…………………………………………………. 8**

**1.1 Introduction**

trendAssist is a business finance and trend prediction program that allows the user to input data over sales made in a day. This data will then be submitted into a database containing financial sales data for every day of the week. It will also allow a user to input data from years past to increase the accuracy of sales predictions; the program will then use this information to generate a new estimation of revenue for that same day of the week in the future and describe requirements needed to hit specific profit margins. There will be two different authorization levels for accounts in the database: standard employee accounts that have access to basic data entry features and owner/manager accounts that have escalated privileges for access to more features about past data. An authorized user (owner/manager) will also be able to determine the optimal staffing needed on any given day to support the expected crowd and maximize profits. The most recent year’s data will be weighted higher than the previous year’s data to maintain accuracy and adjust for recent trends. The configuration items to be covered are as follows: SRS, Sprint backlog, Sprint review, Team member report, Main line, Code line, Release line, MySQL, and Java.

**2.1 Organization**

The software configuration is handled by all members of the project, with specific tools. Responsibilities are shared between all team members.

* The software configuration manager [SCM]
* The project manager [PM]
* The scrum manager [SM]
* 3 developers [D]

These roles change at the end of each scrum sprint.

**2.2 Activities and responsibilities**

**When setting up the project**

* Identify the configuration items [SCM]
* Install the bug repository tool and set up the database [SCM]
* Install the software configuration repository tool and set up the database [SCM]
* Manage and structure the reference space [SCM]
* Define the configuration processes [SCM]

**During the project lifecycle**

* Export components for modification, test or delivery [SCM]
* Set under control validated components [SCM]
* Create version, write version delivery document [SCM]
* Approve reference configurations [PM]
* Verify version to be delivered and authorize deliveries [PM]
* Backup spaces [SCM]
* Do configuration audits [PM]
* Inspect configuration records [PM]
* Archive reference version [SCM]

**Management activities**

* Manage versions and archives [SCM]
* Manage configuration records [SCM]
* Produce reports and statistics [SCM]
* Manage reference space and its access control list [SCM]
* Manage spaces backup and archive media [SCM]
* Manage quality reports [PM]

**2.3 Decisions process and responsibilities**

At the end of an activity of the project

* Do a configuration freeze [SCM]
* Present a configuration state of the components impacted by the activity [PM]
* Present a documentation state of the components impacted by the activity [SCM]

Configuration management process audit

* Do the configuration management process audit [PM]
* Present the records of the configuration management process [SCM]
* Present the quality records of the configuration management process [PM]
* Present the records of the documentation management process [SCM]

**3.1 Configuration identification**

**3.1.1 Identify configuration items**

* SRS
* Sprint backlog
* Sprint review
* Team member report
* Main line
* Code line
* Release line
* mySQL
* Java

**3.1.2 Name configuration items**

* SRS.x.x
* Sprint backlog.x.x
* Sprint review.x.x
* Team member report.x.x
* Main line.x.x
* Code line.x.x
* Release line.x.x
* mySQL.x.x
* Java.x.x

**3.1.3 Acquiring configuration items**

* These configuration items are software, configuration changes will be nonphysical.

**3.2 Configuration control**

**3.2.1 Requesting changes**

* Changes can be requested by customer via E-mail
* Changes can be requested by [D] by internal memo

**3.2.2 Evaluating changes**

* Changes requested will be reviewed by [SCM] then sent to [PM]
* Changes requested will be reviewed by [PM] then send to [D]

**3.2.3 Approving or disapproving changes**

* Changes will be checked for feasibility
* Changes will be evaluated for cost
* Changes will be evaluated for priority

**3.2.4 Implementing Changes**

* Change requests are submitted by the project manager according to the problem resolution process
* When a change request is accepted by the project manager/product manager, a branch is created by the [SCM]
* Branch Identification will be trendAssist.[CustomerID].x

**3.3 Configuration status accounting**

**3.3.1 Metrics to be tracked and reported**

* Number of customers using version x.x.x of trendAssist
* GitHub version control and metrics tracking
* Scrum management
* Document: The modification sheet number identifies the origin of the modification. The modified paragraphs in the document are identified, if possible, by revision marks
* Source file: The software configuration management tool records, for each source file or group of source files, a comment where it describes the modification
* Configuration item: The Version Delivery Description of the article identifies the modification sheet included in the current version
* The modification sheet will describe the modifications done to the components with enough precision to identify the modified parts

**3.3.2 Storage and access control of status data**

* GitHub is used for storage control of the trendAssist software and configuration
* Consumer data will be stored locally on the customer’s device unless otherwise requested, to protect customer data.
* The records are stored in a configuration folder, which contains:
  + The requests sorted by record number
  + The software documents
  + The configuration states, sorted chronologically

**3.4 Configuration evaluation and review**

**3.4.1 Auditing**

* Audit of CI’s by the [PM] and the [SCM] if any configuration changes need to be made.
* Baseline audit, functional configuration audit, software configuration audit.
* Audits identify traceability of product changes to the authorization for the change, and ensure that only authorized changes are implemented in the controlled system.
* Auditing documents authenticate the current configuration of the system and procedures.
* If any discrepancies are noted, they will be corrected and signed off by management and archived in the central repository.

**3.4.2 Definitions**

* The objective of reconfiguration is to change the system to be more accommodating to the customer.
* The schedule will be defined by the [SCM] based on the customer’s needs.
* The PM will work with the [SCM] to audit the changes made to the code line before anything is committed to the base line.
* The [PM], [SCM], [SM] and [D] will meet to discuss the changes made before deciding if/when any version becomes a release line.

**3.5 Interface control**

* Changes made to any CI’s to interface with items outside of the scope of this document after the PM approves will have the same audit procedure as any other change or revision.

**3.6 vendor control**

* Product redistribution outside of our team is **STRICTLY** prohibited.

**3.7 Release Management and Delivery**

* As [D] decides to bring a potential change to the PM to commit from the code line to the base line.
* The changes can now be compiled, and the executable created.
* If the [PM] decides to commit the change to the main line, then the [SCM] will make the changes to the CI’s.
* If the change passes the audit, then it can be committed to the release line and recompiled to an executable.
* Distribution of the product will be handled by the [PM] sending an email to the customer.

**4.1 Sequence and coordination of SCM activities**

**4.1.1 Configuration Baseline**

* The baseline is managed and controlled by the [PM] and solely at his discretion is it pushed to or modified.

**4.1.2 Implementation of change control procedures**

* The SCM tool should automatically generate the unique identifier that will be used to track the CRs, such as ClearQuest. The automatic numbering and assignment of the electronic change request form is handled within SCM tool. This unique number will be tracked in the SCM tools to ensure the change is traceable from approval to implementation and in the case of disapproval; the unique number will remain with the change record showing the disposition.
* The Program/Project Manager must seek a waiver for use of non-approved SCM tools.
* The SCM Manager must list the actual SCM tool and waiver status (if applicable) of tools used to perform development of SCM activities for the system.
  + 1. **Audit Scheduling**
* A multi-disciplined technical review to ensure that the build under review is ready to proceed into formal test. The audit ensures that the build was completed, discusses any issues the build may have had, and confirms that the test environment is ready for the new build to be deployed. This ensures that the test team is aware of any environment changes and/or any additional or missing functionality of the build and they are ready to accept the build.

**5.1 SCM resources**

**5.1.1 Infrastructure**

* The system shall be portable to increase accessibility.
* The system shall be compatible with multiple OS’s to increase accessibility.
* The system shall be able to run on devices with at least Windows XP installed.
* The system shall be able to run on devices with at least Mac OS X 10.8 installed.
* The system shall be able to run on devices that have at least 1GB of memory.
* The system shall be able to run on devices that have at least a 1GHz processor.
* The system shall keep the database backed up to a server in case of catastrophic failure.
* The system shall restore a past copy of the database from the server if need be.
* The system shall reconstruct the database by redoing the operations of committed transactions up to the time of failure.
* The business owner’s username and password will be manually inputted into the account database by the development team. For security purposes, only the business owner can create other accounts.
* The business owner will be able to create sub-accounts with limited access to the data and provide account information to its employees.
* This system shall be available 95% of the time under normal use in under 5 seconds.
* The space required for this system will be kept under 1GB so that trendAssist can be used on older devices with limited storage space.

**5.1.2 Tools**

* Intellij IDEA: a Java IDE with plugins such as JavaFX and SceneBuilder to help build the user interfaces for the software. This is widely used with several resources available for troubleshooting.
* Eclipse: a Java IDE, one of the most widely used Java development tools there are. (This will not be configured to fully run the application, just strictly back end files)
* MySQL Workbench: a GUI version of MySQL database to modify and configure the databases used for this application.
* GitHub: a repository website that makes version control manageable with predefined main line, code line and release line settings.
* KanBan (via ZenHub): an issue resolution platform that is used to describe and keep track of problems that arise during development.

**5.1.3 Equipment**

* **Not applicable**

**5.1.4 Training**

* One of the [D] in our group will be training an employee after installation of the system.

**6.1 SCM Maintenance**

**6.1.1 Plan Changes**

* Maintaining and monitoring the plan is the responsibility of the [SCM].
* Updates are to be performed when the customer requirements change or the [PM] / [SCM] request a change to be made.
* Changes made to the plan are proposed by anyone on the team then sent to the [SCM] to be approved before going to the [PM] for approval.
* Changes then are sent from the [PM] back to the [SCM] to be implemented in this document.

**SCMP version log.**

**REV 0.0.1 3/2/2018**

**REV 1.0.0 3/6/2018**

**REV 1.0.1 3/7/2018**

**REV 1.1.0 3/8/2018**

**REV 2.0.0 4/3/2018**

**REV 2.0.1 4/5/2018**