*Florida International University*

*School of Computing and Information Sciences*

CIS 4911 - Senior Capstone Project

Feature Document

User Story #1153

# VIP4 Version Control

**Team Member:**

Joseph Gonzalez

**Product Owner(s)**:

Masoud Sadjadi

**Mentor(s)**:

Mohsen Taheri

**Instructor**: Masoud Sadjadi

# **User Story - VIP4 Version Control**

* As a developer, I would like to make sure all members of the team are using the right code for Implementation

## **Use Case**

**Use Case ID:** (VIP-1153)

**Details:** Adding and organizing an Integration Branch to the shared VIP4 Github repository

**Actor:** Developers

**Pre-Condition:**

* Developer is a member of the VIP4 Project
* Developer has access to the shared Github Repository
* An Integration Branch has been created on Github

**Description:**

1. Use case begins when a developer’s feature has been considered “done” by the project owner.
2. The developer then proceeds to upload any code changes they’ve made to their personal branch on the VIP4 github (<https://github.com/FIU-SCIS-Senior-Projects/Vertically-Integrated-Projects-VIP-Ver-4.0>)
3. Developers then proceed to add their changes to the features changelog in the shared Google Drive (<https://drive.google.com/drive/u/0/folders/0BwRxNPGpQTNydGdmQkFSX1o0T1E>)
4. The developer in charge of the Integration Branch applies each developer's changes to the original code
5. The integration developer then tests all updates on their localhost to ensure no issues occurred between the features.
6. These changes are then uploaded to the Integration Branch on Github

**Post-Conditions:**

1. All developers now have access to an updated version of the sites code
2. This code must be re-updated after each sprint

**Alternative Course of Action**

N/A

**Exceptions:**

1. If a developer's features were not approved by the product owner, they will not be added to the integration branch

**Related Use Cases:**

* Maintaining a personal Changelog

**Decision Support:**

**Frequency:** Medium: Updates occur every 2 weeks.

**Criticality:** Medium: Updates needed to make sure future features aren’t missing any portions of the site.

**Risk:** Medium. Mistakes could cause update losses between developers code.

**Constraints:**

1. **Usability:** 
   1. Requires experience in Github
   2. One Developer must take on extra regular work
2. **Reliability:** 
   1. Github has an upload contraint of 25MB or 100 files at a time, that can be a hinderance on updating the site. Can be bypassed by using Git from the command line.
   2. Availability – Every other week.
3. **Performance:** 
   1. Downloading the contents can take several minutes, and uploading files can take some time as well.
4. **Supportability:**
   1. All modern web browsers should be supported.

**Modification History:**

**Owner:** Joseph Gonzalez

**Initiation date:** 9/26/2016

**Date last modified:** 9/29/2016

**Use Case Diagram**

Add simple Diagram for dev control

**Sequence Diagram**

N/A

**Class Diagram**

N/A

**Unit & Integration Tests:**

**Test Case 1 (Sunny Day)**

**ID: VIP-SD-1153-01**

**Purpose**

* Ensure the new features on the site are working as intended.

**Precondition**

* Developer must be running the latest version of the site

**Input**

* Developer goes through all the features use cases

**Expected Result**

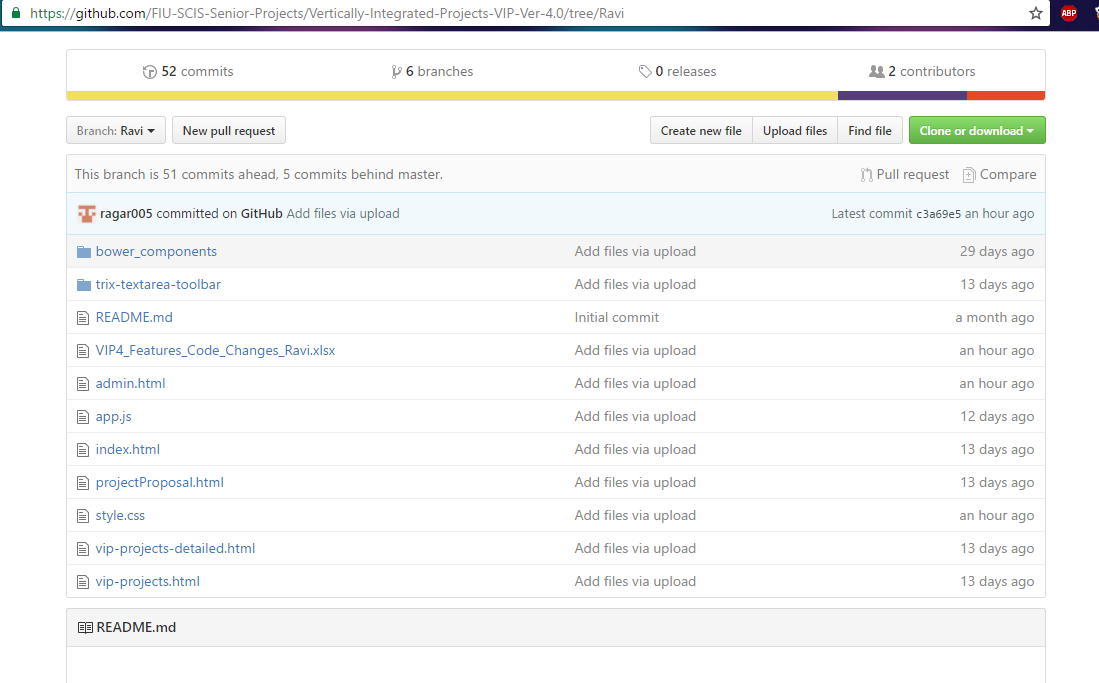
* The features run as intended

**Actual Result**

* Same as expected.

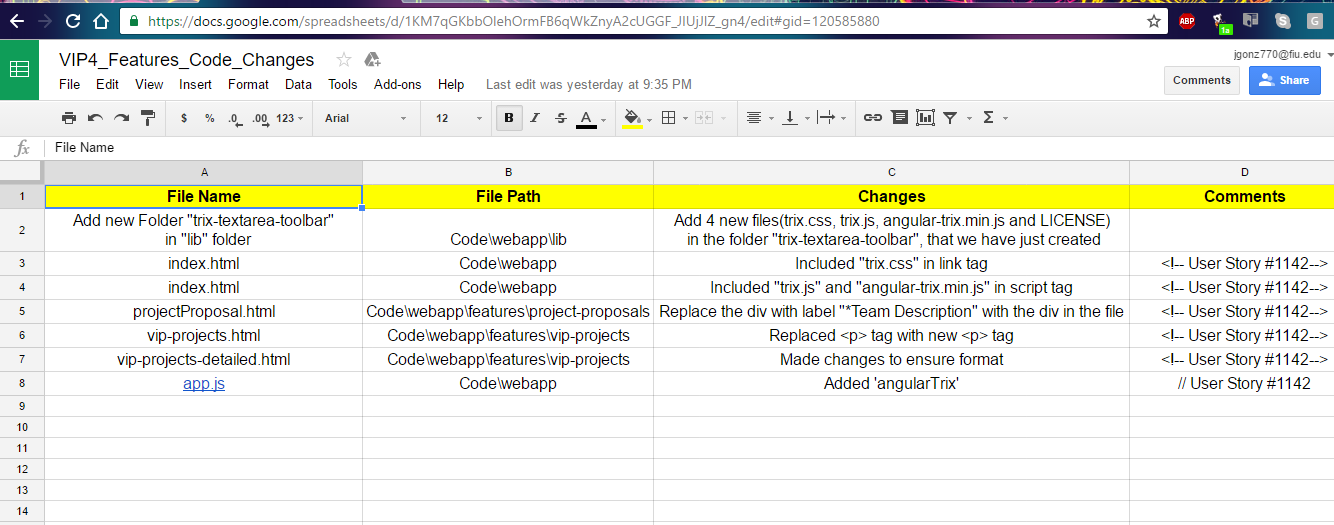
# User Guide

So, in the shared github, every developer must create and maintain their own branch:

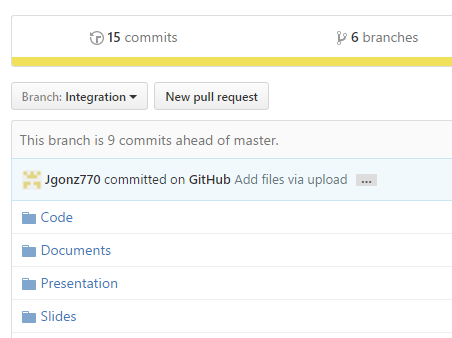


It is here where they upload any code they’ve altered or created for their features.

Developers will also be responsible for maintaining a changelog in the shared Google Drive Repository, where they state what changes they’ve made, to which code, on which lines, and why



When the Product Owner approves features at the end of the sprint, Developers update their Github branch, and Changelog, and the Integration Developer works on updating the integration branch with their changes



Once the branch is updated, the Integration Developer must perform some Integration tests, to ensure the new features work cohesively with the existing system, so that they can be passed forward to the Dev, and eventually Live servers of the site.