* Risks and mitigations

**Project Execution Plan**

0.0.1 – November 23, 2016

# 1.0 Introduction

This document outlines the project execution plan for the subreddit recommender system deliverable being built for MAI-CI. Given the complexity of the project, and a sizeable team of 4 members, some project management techniques are needed for project development. Techniques adapted for this project include: Version Control (git), Code Reviews, Scheduling and Task Assignments (trello). This document also includes a high-level function specification for the final product.

# 2.0 High Level Functional Specification

The goal of this project is to utilize neural computational techniques to develop a recommender system capable of suggesting subreddits of interest to a user based on their historical public interaction with reddit.

## 2.1 Tooling

**Version Control**

* Git via github

**Scheduling and Task Assignment**

* Trello

**Scripting and Data-Munging**

* Python

**Model Development**

* Python – TensorFlow

**User Interface – (Final UI TBD)**

* Python – Flask webapp

Or

* Python – PRAW Reddit Bot interface

## 2.2 Data Gathering

Most of a user’s interactions with reddit are publicly available through reddit’s API. A python wrapper for reddit’s API, PRAW, will be utilized to collect the user data required for this project. The data available from PRAW includes:

* User Comment Text
* Comment Timestamp
* Comment Subreddit
* Comment Submission Title

From this data, predictive features can be developed to train the model in identifying subreddits of interest to the user.

## 2.3 Data Cleaning and Exploration

Once the data has been collected, an EDA will be performed to identify missing and anomalous data to ensure data quality, and trends will be visualized to help inform feature engineering and model development.

## 2.4 Feature Engineering

Feature Engineering will be required to transform the data into useful and predictive formats as the input layer to the neural model to assist in its ability to correctly recommend subreddits. These features may include:

* NLP features from comment body and submission titles
* Network features of subreddit communities and relationships
* Temporal features from user subreddit interaction timestamps
* Macro Reddit Community features, such as subreddit popularity or trending
* External data sources, such as Pocket integration for article/subreddit discovery

## 2.5 Model Development and Validation

It is proposed to utilize the TensorFlow package for python to implement the neural model for this project. Multiple neural models may be developed and tuned to compare performance, including feedforward Neural Networks and Recurrent Neural Networks. In-depth research about the available methods will be required to understand and implement them for the specific dataset of this project.

The models are trained by extracting a subset of each user’s subreddits as the training labels for the model. These models are validated by holding-out a portion of the dataset to be utilized as the testing set to apply the models and compare results.

## 2.6 Product User Interface -TBD

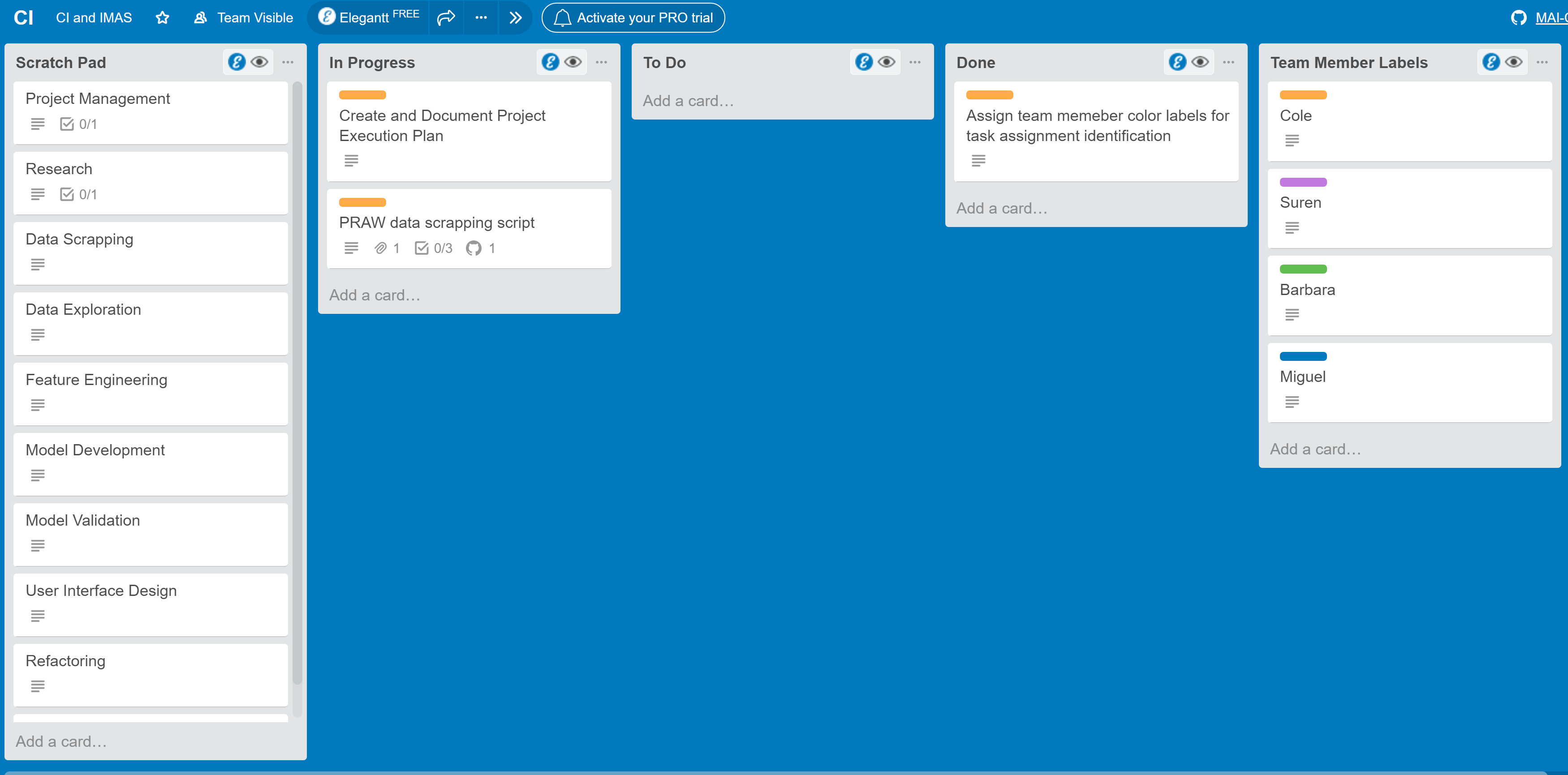
In order to allow users to utilize the recommender system, a user interface is required for them to interact with it. There are currently 2 options for interfacing with reddit users: through a reddit bot or through a custom webapp. Once the model is finalized and the team has a better understanding of its capabilities, a detailed function specification for the final user interface will be developed.

# 3.0 Project Management

This section outlines the processes that will be utilized to ensure successful execution of the proposed project

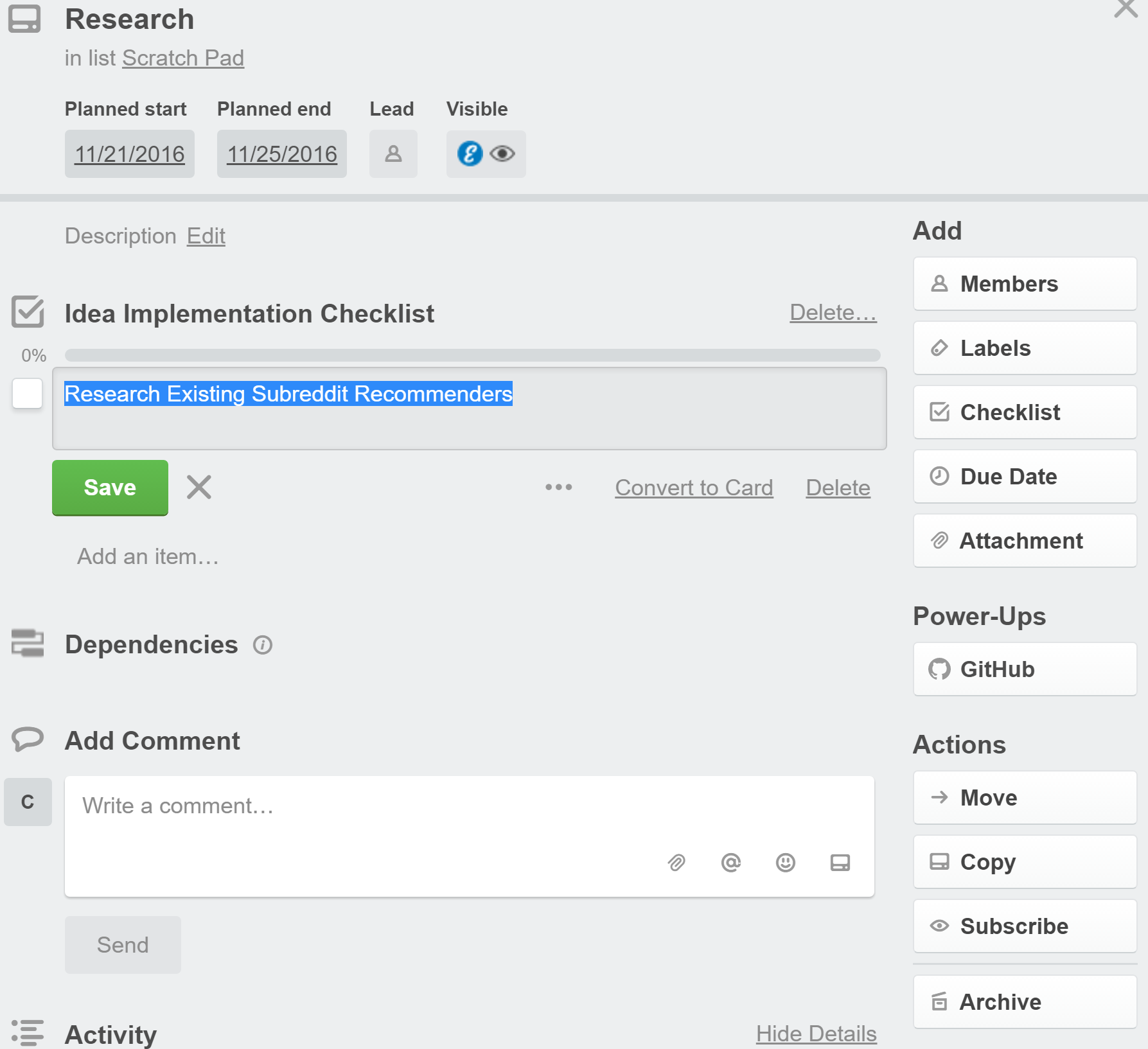
## 3.1 Team Scheduling and Task Assignment - Trello

The project management tool, Trello, will be utilized to create, schedule and assign project tasks to team members. This tool provides a shared interface for creating categorized lists of tasks and has tools for creating schedules and integrations with github.



There are 4 main trello lists utilized for project management:

**Scratch Pad** – This is the idea generation list, where any team member can add an idea for the project to be documented and potentially executed. Ideas are added by selecting the general task group it belongs to, and adding the idea to the Idea Implementation Checklist within the general task group. Ideas can then be promoted to individual task cards for implementation by selecting the “Convert to Card” option on the listed task. This ensures that all ideas are captured, but only those requiring implementation become promoted into the workflow.



**To Do –** This list tracks all of the tasks that require implementation. Any team member can promote a Scratch Pad idea to the “To Do” list by making the idea an individual card and dragging it into the “To Do” list.

**In Progress –** This list tracks the task currently being executed by a team member. Tasks are color labelled by the team member actively executing the task. Any team member can promote a task from the “To Do” list into the “In Progress” list by assigning the task to themselves, scheduling the task and attaching the task development github branch to the task.

**Done –** This list tracks all of the tasks successfully completed for the project. In order to promote a task from “In Progress” to “Done”, the following subtasks need to be completed.

* Code commenting and documentation
* Successful merge into Development
* Successful merge into Master

## 3.2 Version Control – GitHub

This project will utilize the GitHub source control infrastructure for project version control, and will implement the small team git workflow suggested by [Pro Git](http://stackoverflow.com/questions/2428722/git-branch-strategy-for-small-dev-team). The workflow includes a master branch for stable production code, a development branch for individual feature merging and individual task branches for development of each unique task. Each task in the “In Progress” trello list is assigned a task branch, and once complete is merged into the development branch to check for any conflicts with other team member feature developments. Once conflicts are resolved, the conflict free development branch can be merged into the master branch.

# 4.0 Workflow and Checklist

This section outlines the full workflow required for any project task.

1. Idea Conceptualization

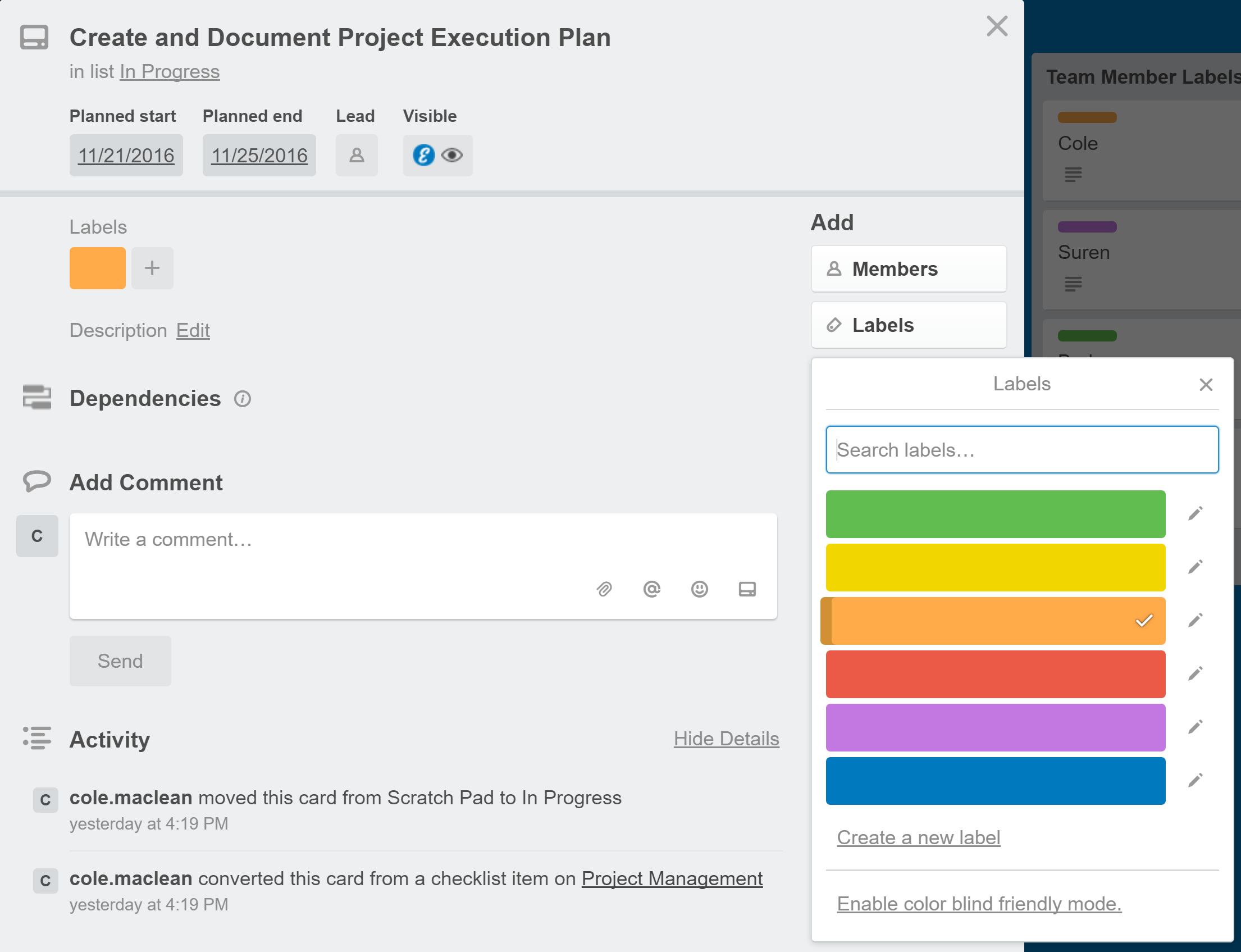
* Select the general task group the idea belongs to
* Add the task to the Idea Implementation Checklist of the selected general task group

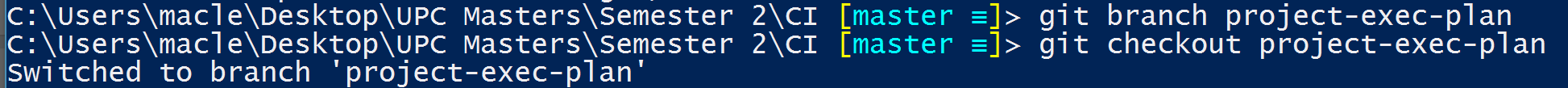
2. Idea to “To Do” Promotion

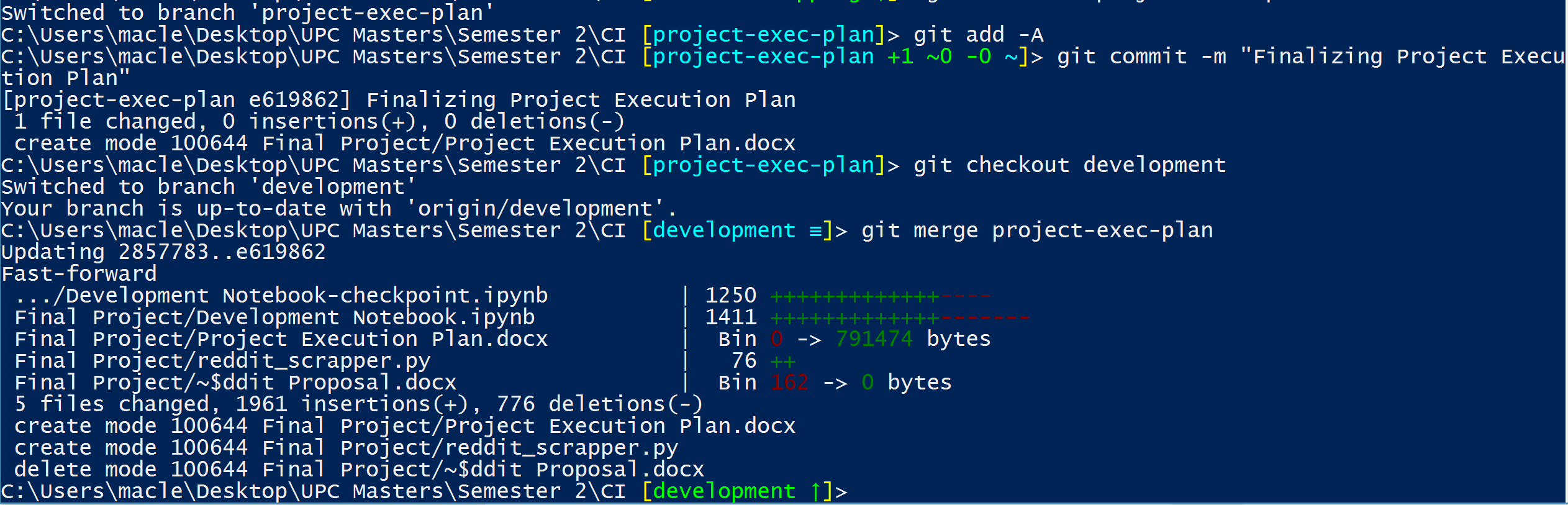
* Identify an idea from a general task group that requires implementation into the project
* Convert the idea into an individual card by selecting the “Convert to Card” option
* Drag the newly created card into the “To Do” list

3. To Do to In Progress Promotion

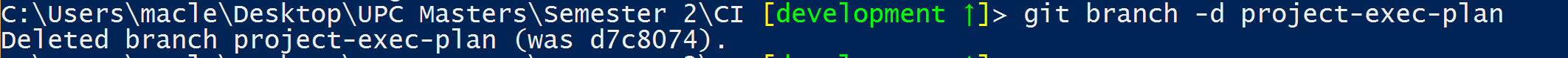
* Identify a task from the “To Do” list that you will implement
* Assign the task to yourself by labelling the card with you team member color



* Create a task development git branch
* Switch to the newly created task development branch and implement and test the project task
* Once complete, merge the task development branch into the Development branch



* Resolve any merge conflicts
* Once conflicts are resolved and merge is successful, delete the old task development branch



* Depending on the stage of development, the team will decide when the development branch is merged with master