

Project Development Manual

HonestWorks

11/24/2020

Team Member – **Abel Marin Jr**

Team Member – **Andrew West**

Team Member – **Fazal Mohammed**

Team Member – **Michael Bottom**

Team Member – **Bilal Chaudhary**

## Table of Contents

Section 1: Requirements Document.....	3
Section 2: System Technical Specs.....	8
Section 3: Design/Architecture.....	10
Section 4: DB Design.....	12
Section 5: Wireframe.....	13
Section 6: Test Design Plans.....	17
Section 7: Use Cases.....	19
Section 8: Project Code Module List.....	20
Section 9: User Manual.....	21
Section 10: Weekly Team Log.....	22
Section 11: Team Work Logs.....	23

## **Section 1: Requirements Document**

### **Introduction**

When working on a project, feedback is a critical component of correcting mistakes, preventing future error, assessing improvements, and optimizing the efficiency of the next project cycle. However, when honest feedback from team members is not given, it hinders the quality of a project cycle, and can impact the efficacy of a project team as well as the overall product; that is where HonestWork can make a difference.

#### **1.1. Purpose of the system**

The purpose of this project is to allow for honest and productive feedback amongst team members without the fear of judgement, retaliation, or nonconstructive critique. Developing a web application that allows users to do this makes the project cycle more effective and permits users to remain nameless whereas doing this in person could take more time and potentially intimidate team members from offering genuine feedback.

#### **1.2. Scope of the system**

The scope of the project is to allow users within the same team to make projects, tasks, and posts commenting on the completed tasks of other members as well as vote on best feedback; this way, all users will be able to critique the work of all other users. Critique is intended to be done on the performance of the task. Aside from this, an assigned administrator will be set as moderator to ensure only useful, relevant, and unbiased feedback is given. This is to prevent users from posting irrelevant or

inaccurate responses to the completed tasks of their peers.

### **1.3. Objectives and success criteria of the project**

- Create an application that allows users to make projects, tasks, and posts reviewing the tasks completed by other members of the same team on the same project.
- Allow users to vote on the best feedback for each task.
- Give an assigned member admin privileges to moderate commentary made from user to user.

### **1.4. Definitions, acronyms, and abbreviations**

GUI - Graphical User Interface: The application's visual representation or interface

SQL- Structured Query Language: Used to communicate with application's database

RAD - Requirements Analysis Document: specifically details what is needed for the successful development of the product

ERD- Entity-Relationship Diagram: a type of flowchart that illustrates how "entities" such as people, objects or concepts relate to each other within a system.

HTML- Hypertext Markup Language: the standard markup language for documents designed to be displayed in a web browser

CSS-Cascading Style Sheets: a style sheet language used for describing the presentation of a document written in a markup language such as HTML

## **1.5. References**

[p://www.cs.fsu.edu/~lacher/courses/COP3331/rad.html](http://www.cs.fsu.edu/~lacher/courses/COP3331/rad.html)

Source of Requirements Analysis Document

[http://www.doit.state.nm.us/project\\_templates.html](http://www.doit.state.nm.us/project_templates.html)

Project Management Tools

## **1.6. Overview**

Feedback has become an essential aspect to any team or project that aims for growth and effective collaboration. Far too often effective feedback is not provided in these situations, and this does not benefit a project in any way, and may in fact serve as an overall negative. There are two main issues with a feedback system that largely exist today. The first is that feedback is not provided at all, and the second is that when this feedback is provided, it proves to be ineffective or not well thought out. Companies such as Netflix have aimed to provide a culture where brutally honest feedback is not only accepted but encouraged. HonestWork aims to develop a system where this culture can be applied throughout a project. This is accomplished by not only providing feedback, but making this feedback as effective as possible.

We are aiming to build a web-based program where users will be divided into admins and team members. The user experience will be similar to existing softwares such as Trello where tasks are divided by what point they are within the process. Admins will create projects with tasks underneath these projects. As these tasks are completed each member of the team can anonymously post their opinion of how they felt said team member completed the task. A point system will be created to add a layer

of gamification to the software. This will help to ensure that all feedback is effective and quality feedback, and not just random thoughts.

### **Current System**

Systems currently exist to help teams manage their projects and split them into tasks to be assigned. None of these systems provide incentive or require the posting of honest feedback. These systems provide for the organization that a team needs to complete a project but they don't allow for the team to improve the project and their skills.

### **Proposed System**

#### **1.7. Overview**

Our system would be similar to the existing systems in design and purpose with a focus on incentivizing and rewarding honest and constructive criticism of their team member's work. This will be done through a like system and required comments by team members on the tasks completed by their peers. These comments will then be voted on by the entire team to assign points for the most valuable feedback. There will be a leaderboard for whoever has received the most votes on their comments to have recognition for their contributions to the project and for making the team better as a whole.

#### **1.8. Functional Requirements**

1.8.1. Be able to create projects

1.8.2. Be able to create tasks within individual sprints

1.8.3. Post anonymous feedback to these sprints once they are completed.

1.8.4. Be able to vote on which piece of feedback was the best. Thus

earning the person who posted this feedback “Honest Points”. The Honest Points will then be displayed on a leaderboard.

1.8.5. Allow an admin to moderate feedback.

## **1.9. Nonfunctional Requirements**

### **Non Functional Requirements**

The nonfunctional requirements section is currently a work in progress and not yet fully fleshed out. This is due to the fact that many of the elements of the project can still be altered and changed as we continue to work on the project.

### **Usability**

HonestWork is designed to be easy to use and manage for both users and administrators. The current proposed system is designed to ensure that users and organizations as a whole are able to produce great work by getting team members to give good, brutally honest feedback.

**Reliability** All user information is stored securely and in a safe environment. All tasks are also going to be diagnosed properly and going to be assigned properly. In other words Each task should be assigned to the correct member and each task should be properly marked by its step in the process such as “do, doing, and done”.

### **Performance**

User actions should provide proper feedback within seconds. The tasks should be created by the admin and then quickly generated by the system. System should also properly diagnose which step of the process the task is in by marking a task that is completed as “done”, marking a task in progress as “doing”, etc. Implementation This system is being implemented using javascript, react framework, and node.js.

## **Interface**

All information and user actions should be clearly marked so both admins and users know how to create and complete tasks respectfully. HonestWorks will operate on web browsers and work on phones such as iPhone and Android devices

## **Section 2: System Technical Specifications**

### **Scope**

This section will detail the models of React.js and node.js that we will use for HonestWorks. HonestWorks will also use HTML and CSS to develop our web application that will be available both on laptops and for mobile users.

### **Audience**

Our main audience is project teams aiming to use feedback in a more effective way so that it serves as actual productive feedback. In terms of skills required, the interface is fairly easy to use and the only skill that is really needed is the ability to navigate through a web application. Some



basic team project skills may be needed in order to properly provide feedback and navigate through the different stages of the project.

### **Document Conventions**

GUI - Graphical User Interface: The application's visual representation or interface

SQL- Structured Query Language: Used to communicate with application's database

RAD - Requirements Analysis Document: specifically details what is needed for the successful development of the product

ERD- Entity-Relationship Diagram: a type of flowchart that illustrates how "entities" such as people, objects or concepts relate to each other within a system.

HTML- Hypertext Markup Language: the standard markup language for documents designed to be displayed in a web browser

CSS-Cascading Style Sheets: a style sheet language used for describing the presentation of a document written in a markup language such as HTML

### **System Overview**

#### **Description**

The purpose of this project is to allow for honest and productive feedback amongst team members without the fear of judgement, retaliation, or nonconstructive critique. Developing a web application that allows users to do this makes the project cycle more effective and permits

users to remain nameless whereas doing this in person could take more time and potentially intimidate team members from offering genuine feedback.

### **Objectives and Success Criteria**

- Create an application that allows users to make projects, tasks, and posts reviewing the tasks completed by other members of the same team on the same project.
- Allow users to vote on the best feedback for each task.
- Give an assigned member admin privileges to moderate commentary made from user to user

## **Section 3: System Design/Architecture**

### **Software Design**

#### **Software Packages**

There will be three main modules which will all be built using react and node with SQL as the database service. These are as follows: Feedback module, Project module, and the Organization module.

#### **Software Module: Organization Module**

Functionality: Create and edit projects, add new people to the organization

User Relations: There will be admins and users. The admins will be the people creating projects and adding new people

**Software Module: Project Module**

Functionality: Create and edit tasks, add new people to the project

User Relations: There will be admins and users. The admins will be the people creating projects and adding new people

**Software Module: Feedback Module**

Functionality: Leave anonymous feedback, be able to give a piece of feedback an HonestPoint, leaderboard displaying the amount of HonestPoints each team member has accrued

User Relations: There will be admins and users. The admins will be in charge of making sure that there is no rude or offensive feedback. The users will be the ones posting the feedback and handing out HonestPoints.

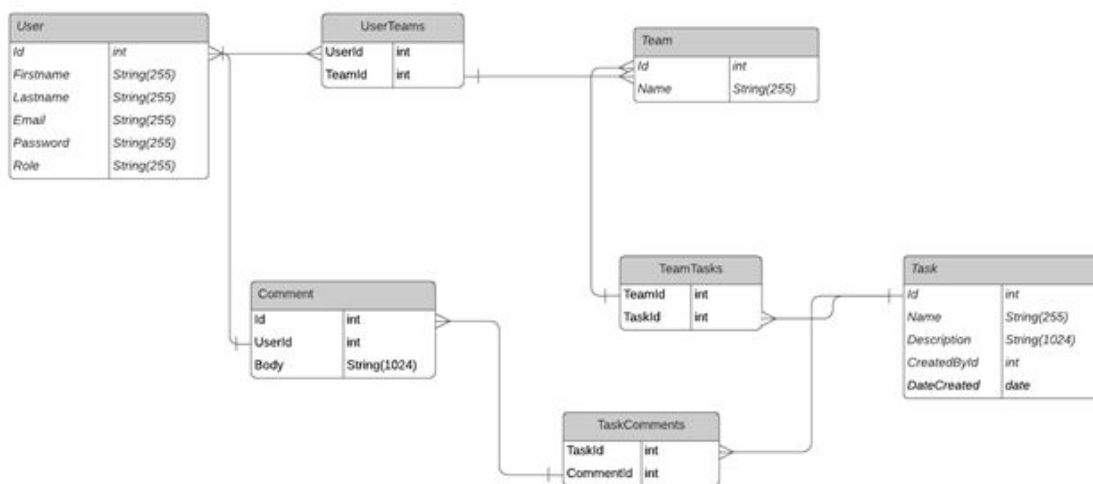
**Software Integration**

The three main modules will essentially work like Russian nesting dolls. With Feedback being contained within project tasks and Projects being contained within Organizations.

## Section 4: DB Design



### Database Design

There will be a table of users. The table will include usernames, among other data. This database will make use of SQL to provide a good experience for users.

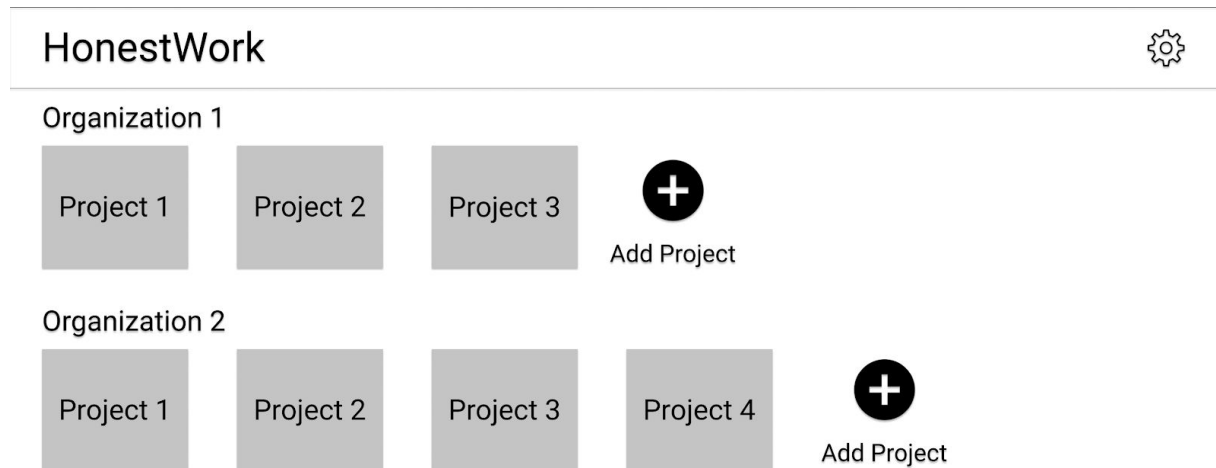


## Section 5: Wireframe

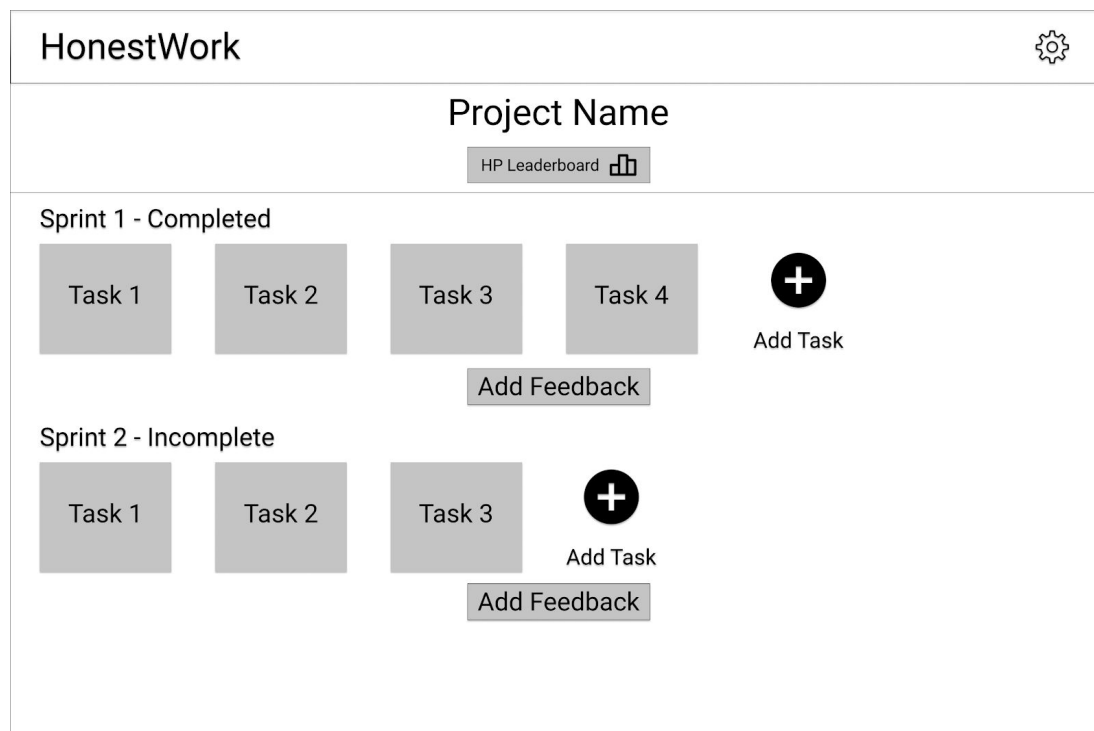
This is what the login page will look like:

HonestWork	
<b>Account</b>	
Change Profile Picture	
Change Password	
<a href="#">Delete Account</a>	

This is the outline of the homepage:



This is the outline of what a person would see after clicking on a project from the homepage:



This is what a task page will look like:

HonestWork

Task Name

Task Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam fringilla ligula vel blandit auctor. Nullam et ligula sed risus tempus rhoncus. Curabitur ac vehicula quam. Duis dictum eros ac tortor viverra pellentesque. Morbi ac quam id neque tristique dignissim. Etiam in pretium felis. Sed sed velit nec dolor vehicula fermentum. Aliquam porta sed nulla at pellentesque. Aenean iaculis ornare massa ut placerat. Curabitur nec convallis sapien, quis efficitur risus. Maecenas consectetur risus risus, sit amet lobortis tortor egestas at. Cras lacinia dictum nulla non facilisis. Curabitur eleifend nulla non lacus blandit eleifend.


Complete Task

This will be the leaderboard for the people who have gained the most honest points:


HonestWork

Honest Point Leaderboard


#1 Team Member



#2 Team Member




#3 Team Member



#1	Team Member Name	HP Score
#2	Team Member Name	HP Score
#3	Team Member Name	HP Score

This is outline for the feedback submission page:

HonestWork



### Honest Feedback


Subject

Feedback

Submit

This is the outline for how the feedback page will look:

HonestWork



### Project Name

Sprint #1

Feedback #1

+HonestPoint

Feedback #2

+HonestPoint

Feedback #3

+HonestPoint



## Section 6: Test Design Plan

### 1.1 User Problem

Companies need a way for team members to give brutally honest feedback.

### 1.2 Goals of the Project

Create a webpage where company team members can give honest constructive feedback.

### 1.3 System Scope

The webpage will provide a page for users to submit posts. These will be organized and displayed appropriately. Other users can view these posts as feedback.

### 1.4 Client, Customer, and Other Stakeholders

The Client - Company employees.

The Customer - Other company employees.

OtherStakeholders - The company that employs the employees.

### 1.5 Users of the Product

Users of the product will be coworkers, cross-department coworkers, and managers.

<Functionality Requirements contained in Part 1>

### 1.6 Usability Requirements

For our ease-of-use requirements, we want users to be able to easily find the posting tools and easily submit posts with as little hassle as possible. For our documentation requirements we want users to have easy to follow and easy to access instructions on submitting posts and what to expect after submitting posts. For safety requirements we want users to have access to a list of policies or regulations that may need to be adhered to.

**1.7 Performance Requirements**

We want to use a cloud service to ensure availability, reliability, capacity, scalability, and the ability to document this. This will also allow us to recover and restore backups in the case of an emergency.

**1.8 Security Requirements**

For user security requirements we have two current roles, user and administrator. They will have a different set of security permissions. The user should only be able to see its own profile and only posts that have been authorized for users to see. The only confidential or private information we plan to store is the user's address, email, etc.

**1.9 Legal Compliance Requirements**

Our current legal notifications are:

You publish your contributions under free licenses.

The service will resist legal requests for user information where reasonably possible.

The service can delete your account without prior notice and without a reason.

Our privacy requirements are that all private user data is protected, and possibly encrypted. We currently have no funding requirements.

## Section 7:Use Cases

### Actors

The actors are mainly employees, team members, and possibly managers.

### Task-level Goal

The goal is for actors to be able to submit their tasks on the app, describing the detailed structure of their projects. And other employees, team members, or managers can give feedback on those projects.

### Priority

High - Method for submitting a project.

Medium - Method for detailing structure of the project.

High - Method for reviewing and critiquing the project.

Medium - Method for aggregating or averaging reviews into a feedback score for a single project.

Medium - Method for actors/users to quickly find projects they have interacted with before.

### Brief

For all interactions the users must register for an account.

**Method for submitting a project:** The user will click a button to start a new project, that leads to a new page. On the new page there will be input boxes that the user will use to input information about the project and the project summary. Then the user will click on a button that submits the information.

**Method for detailing structure of the project:** The user will be able to go back and edit the input boxes that provide information about the project to add more detail as necessary.

**Method for reviewing and critiquing the project:** There will be a button on a project page that users will see that, when clicked on, will lead to a new page. On the new page users will be able to add feedback in input boxes. There will be a button on the page to submit the info and the info will be posted for viewing.

**Method for aggregating or averaging reviews into a score for a single project:** On the server side code, user review scores will be averaged and a single score will be provided to be used as an overall feedback score for the user-submitted project.

**Method for actors/users to quickly find projects they have interacted with before:** Each user will have a profile page link on their main page. When they click on their profile page link and go to their profile the user will see links to the past projects that they have interacted with.

## **Section 8: Project Code Module List:**

- To support the application basic web development skills are necessary. This includes HTML, CSS and an understanding of how to host and secure a website.
- In addition to the basic web skills, competency in javascript, the React framework and Node.js is required.
- Finally, an understanding of database management using SQL is the last major tool that must be learned to support the project.

- Also used Heroku to put together front end of project
- Used Github repository to store back end code

## **Section 9: User Manual**

### **User Roles:**

#### Team Member:

- Create and finish task within the projects
- After each task is completed, each member will be required to give honest feedback on the quality of the completed task

#### Administrator:

- Create projects
- Moderate discussions between the different team members

#### UX:

- Users will be divided into admins and team members. User experience will be very similar to trello where tasks are divided by what point they are within the process, i.e. subdivided beneath “posts” - to do, doing, done, and feedback.
- Admins can create projects which will be topics, and below that there will be tasks generated by team members. After a task has been completed by a team member, each member of the team can anonymously post their opinion of how they felt said team member completed the task.
- There will be a point system included to rate different criteria of the way the task was completed within the post.

**Section 10:Weekly Team Log**

**Week 1:** Discussed potential project plans. Met and sifted through different project options while weighing pros and cons of each option.

**Week 2:** Decided to do our project on HonestWorks. Discussed how it can be used as an effective feedback tool with elements of gamification.

**Week 3:** Put together Requirements Analysis Document and worked through all functional and non-functional requirements of the system.

**Week 4:** Started discussing how to code the front and back end of the project. Decided to use GitHub and Heroku

**Week 5:** Put together a document with System Architecture and Use Cases. Discussed how the system will be put together and used.

**Week 6:** Put together a mockup of the front end on Heroku while continuing to work on the back end on Github.

**Week 7:** Completed the System Plan which included putting together Wireframe. Also displayed navigation of the interface.

**Week 8:** Finished up coding the front end of the project and began working on how to connect front end to back end.

**Week 9:** Worked through building the database and figuring how to connect this database to the back end and then the front end of the project.

**Week 10:** Finished up back end coding and connected the back end coding to front end coding.

### **Section 11: Team Work Log**

<https://github.com/abelmarin/honest-work>

<https://dashboard.heroku.com/apps/honestwork-app>

<https://honestwork-app.herokuapp.com/>