

FETCH BOT DEVELOPMENT STANDARDS

AI based robot with image processing.

<https://cshadd.github.io/fetch-bot/>

In use since 3/13/18.

Contents

1. Overview	3
1.1. Scope	3
1.2. Purpose	3
2. Project Management	3
2.1. Model	3
2.1.1. Model Elements	3
2.1.2. Model Organization	3
2.2. Meetings	4
2.3. Perfection.....	4

1. Overview

Clause that defines the overview of the standards of development for Fetch Bot.

1.1. Scope

Development process for Fetch Bot.

1.2. Purpose

The point of this document is for the developers of this project to learn the model of our project management and the tools we use. Furthermore, it is important for our stakeholders to understand our process cycle.

2. Project Management

IEEE Standards IEEE Std. 1074-1997 recommends we have a software life cycle model and IEEE Std. 1490-1998 recommends we have a project management body. When we use terms related to the software life cycle model and the project management body, we refer to these IEEE Std. documents for reference. Most importantly we understand that these are abstract and we may choose to partially implement these ideas.

2.1. Model

Researching for a specific model began during the early stages of the project. We were using a loosely based Scrum model. After a few weeks in, we have transitioned to a hybrid Agile Scrum model as our main provider, GitHub, did not provide the necessary tools for time management within the scope of our project.

We used <https://www.zenhub.com/>

Zenhub Guide: https://www.zenhub.com/guides?utm_source=Dashboard

2.1.1. Model Elements

- User Stories – Also known as Issues, these are tasks that the developers acknowledge must be completed. There are two types of Stories:
 - Stories – Stories that can be grouped together into an Epic.
 - Standalone Stories – Stories that cannot be grouped together into an Epic and are completed separately from other Stories.
- Epic – Also known as “super” Issues, these are collections of User Stories the developers acknowledge must be completed. Every Story within an Epic must be completed for the Epic to be completed.
- Sprint – Also known as a Milestone, this is a deadline for a set of User Stories that need to be completed.
- Release – Not to be confused with product release, an Agile Scrum “Release” is a deadline for a set of Epics and Standalone Stories that need to be completed.
- Points – Priority labels given to estimate the priority and/or completion of a Story or Epic.

2.1.2. Model Organization

Below is a table for the organization of Model Elements. First Column is the Type, First Row is the Need.

Take note that if something says NEEDS ALL, that refers to all the elements within the deadline of the Type.

	User Story	User Story (Standalone)	Epic	Sprint	Release	Points
User Story						NEEDS
User Story (Standalone)						NEEDS
Epic	NEEDS					NEEDS
Sprint	NEEDS ALL	NEEDS ALL				
Release	NEEDS ALL	NEEDS ALL	NEEDS ALL			
Points						

2.2. Meetings

We shall have our Daily Scrum meeting once a day if possible. This meeting is meant to look at the progress and relationships between the developers of the project. It is a great opportunity to bring up topics to encourage other users or to bring up any concerns. During that time, team members share what they worked on the prior day, will work on that day, and identify any impediments to progress.

We shall have our Sprint Planning Meetings before every new sprint. This is where we assign a sprint goal and the sprint backlog, or the User Stories that we wish to develop from the product backlog. We will also assign priorities and users to each User Story to complete.

At the end of a sprint, the team conducts a Sprint Review during which the team demonstrates the new functionality to any stakeholder who wishes to provide feedback that could influence the next sprint.

Another activity is the sprint retrospective at the end of each sprint. The whole team participates in this meeting. The meeting is an opportunity to reflect on the sprint that has ended, and identify opportunities to improve.

These meetings are not final and may change.

2.3. Perfection

We are only human, if we were perfect we would be able to complete this project in one day. Therefore, we keep these Standards as a reference for development and may or may not fully follow it as time goes on. These standards are subject to change. We are using a hybrid version of Agile Scrum.