# Gulp

21/2/2020

## Overview

This pill is designed to increase the knowledge in the technology of Gulp. Gulp is a tool designed by NodeJs to automate certain tasks such as compiling different languages, minifying code, auto-refreshing the browser when you modify your code, validating syntax, compressing files, creating custom automations etc. It's often used to do front end tasks like:

- Spinning up a web server
- Reloading the browser automatically whenever a file is saved
- Using preprocessors like Sass or LESS
- Optimizing assets like CSS, JavaScript, and images

Tools like Gulp are often referred to as "build tools" because they are tools for running the tasks for building a website. The two most popular build tools out there right now are Gulp and Grunt. The main difference is how you configure a workflow with them. Gulp configurations tend to be much shorter and simpler when compared with Grunt. Gulp also tends to run faster.

## Objectives of project

- Understand what Gulp is and know its basics
- To be able to deploy a web project that uses gulp to generate various file outputs based on small plugins
- Understand what Gulp plugins are and how to use them

- Understand how a project is configured for different execution environments

- Have tools to perform various automations on a given project quickly and easily

## Table of Contents

## Project requirements

- Requirements documentation.
- List of tasks to be performed.
    - Priority of each task
    - Title and description of each of them
    - Difficulty level
    - Estimated time for each task.
- Record of incidents that were detected during the execution of the pill.
- Schedule or Calendar of the project. (Choose the one you consider most appropriate given the size of the project)
- Quality metrics (what points do you consider important to measure the success of the project)
- Documentation about the git WORKFLOW you are going to use
- Documentation about the tools used in the project

- Record of lessons learned

## Risk Management

Every project has risks. These risks must be taken into account to improve the workflow of the project. Managing these risks is important for due completion of the project. Unchecked risks could not only lead to  hamper of project deliverance but also a badly executed project. The risks hence associated with the project are documented as follows:

| Risk | Risk level |
|---|---|
| Unfamiliarity with  NodeJs | Low |
| Unfamiliarity with  Gulp | Medium to Low |
| Health and computer issues | High |
| Delivering in time | Medium |
| Completing all the project requirements | Medium |

## Task Management

The following are the tasks which needed to be accomplished for the completion of the project. The tasks have been divided according to the project specifics. Each task has been clearly defined and their priority as well as their difficulty and time needed. Difficulty level is explained on a scale of 1 to 5 ( 5 being the most difficult ). Priority is explained on the level of 1 to 5 ( again 5 the highest parameter being the most prioritized work ).

| Task | Priority | Description | Difficulty | Time |
|---|---|---|---|---|
| Read the description of the project | 5 | This task involves complete understanding of requirements for the project | 1 | 30 mins |
| Create git repo | 5 | Creating of git repository for project execution and delivery | 1 | 2 mins |
| Create the directory structure | 4 | Creating the necessary files, folders and subfolders for this project. | 1 | 2 mins |
| Install Node and NPM | 4 | Install Node and NPM if not installed previously | 2 | 4 mins |
| Install Gulp | 4 | Install Gulp locally and Globally and understand how to use its | 1 | 2 mins |
| Create gulpfile.js | 3 | Create the main gulp file | 1 | 1-2 mins |
| Load the modules required | 2 | Load the required modules using npm install and require | 2 | 5-10 mins |
| Hello world | 3 | Write the function to print out ‘hi’ | 2 | 10 mins |
| Html, sass and js tasks | 2 | Run gulp tasks for html, sass and js | 3 | 0:30-1:00 hr |
| Pipes and Glob | 3 | Understand what are pipes and globs and use them in the project | 4 | 1-2 hr |
| Prefix for sass | 1 | Use prefix for sass compiled files as ‘min.css’ | 3 | 10-20 mins |
| Create dist for production version | 1 | Create the dist folder for compiling production version of the codes | 1 | 1 min |
| Create production version | 3 | Create production version | 4 | 0:30-1:00 hr |

|  |  |  |  |  |
|---|---|---|---|---|
| Use browserSync | 1 | Use browser sync instead to keep watching changes in css and change automatically in localhost render | 4 | 30 mins |
| Use inject | 1 | Use inject to inject css and js in html | 4 | 1-2 hr |
| Documentation | 1 | Write the documentation for this project | 1 | 1-2 hr |

Defining this part is crucial to the development of the project. It is important to make a good analysis of the situation to organize the project in a good way.

## Chronogram

This shows the time-line it took for the project to finish and also the tasks that were achieved during those timeline.

The tasks have been mentioned in the left axis

| **Tasks** | Thur 1000 hrs | Thur 1100 hrs | Thur 1200 hrs | Thur 1300 hrs | Thur 1400 hrs | Thur 1500 hrs | Thur 1600 hrs | Thur 1700 hrs |
|---|---|---|---|---|---|---|---|---|
| **Read project description** | **X** |  |  |  |  |  |  |  |
| **Git Repo** | **X** |  |  |  |  |  |  |  |
| **Create files** | **X** |  |  |  |  |  |  |  |
| **Install Node and NPM** | **X** |  |  |  |  |  |  |  |
| **Install Gulp** | **X** |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|
| Load required modules |  | X |  |  |  |  |  |  |
| Create gulpfile.js |  | X |  |  |  |  |  |  |
| Hello world |  |  | X |  |  |  |  |  |
| Html, sass and js tasks |  |  | X | X | X |  |  |  |
| Pipes and glob |  |  |  | X |  |  |  |  |
| Prefix for sass |  |  |  |  | X |  |  |  |
| Create dist version |  |  |  |  | X | X |  |  |
| Browsersync and inject |  |  |  |  | X | X | X |  |
| Documentation |  |  |  |  |  |  | X | X |

## Git Workflow

For this project, all commits we'll be pushed directly to the Master branch. All commits will use a descriptive message, so that the admin and other internet user can follow through the processes.
For more information go to :
https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow

## Incidents

- None

## Technologies used

For this project, we will use the following technologies:

- HTML
- NodeJs
- Npm
- Gulp
- JS
- SASS