Report

Assignment 1

- Personal planning, Vision and Project Plan

31 Jan, 2017

Semester: Spring 2017
Course: Software Technology
Author: Sarpreet Singh Buttar

Contents

Task 1 Personal Planning

Purpose

The purpose of this task is achieve the following objectives:

1. Create a list of books and a method to get them.
2. Convert the created objects into JSON objects.
3. Answer the request in the web browser in order to display the JSON objects on the screen.

Goal

The goal of this task is to present a list of books as a JSON object (an associative array) for the user when the client requests it using http://localhost:9090/api/books/ to the server.

Resources

Workers		
Role	**Resources Recommended**	**Responsibilities**
Implementer	1	All implementation and documentation

Tools	
Tool	**Purpose**
Atom	Implementation
Vagrant	Code execution

Risks

Problem	**Effect**	**Solution**
I'm the only person working on this assignment and I might become unable to work on it because I have one part time work and two other courses to study.	The assignment will suffer greatly and ultimately might become a failure.	None
I've never programmed in editor.	It might be more time-consuming for me to implement.	Take this into consideration when planning the schedule.
I've never worked with vagrant.	It might show some error which I am not aware of.	Take this into consideration when planning the schedule.

For achieving the above objectives, this task is further divided into three substasks. Each substask follow the *Plan, Implementation and Reflect* pattern.

Subtask A Books

Plan

Objective The objective of this subtask is to create a list of books and a method to get them. Also, print the created books in the terminal.

Requirements The requirement of this subtask is to verify that the list of books is printed out in the terminal(where vagrant is run) when calling the URL http://localhost:9090/api/books.

Strategy Below is the strategy for achieving the requirements:

1. Read the given source code and API.
2. Implement a *book* class in the *models* package. This class must have six fields named as id, title, author, genre, publishDate and description for presenting a *book*.
3. Create some fictive objects of type *book* in the method *getBooks* that is available in *GetBooksResource* class in the *resources* package.
4. Use *println* method to print each book in the terminal.

Reflection

All the implementation went according to the plan and program run successfully in first attempt. I thought that implementation will only take 15 minutes but it took more because I am using editor rather than IDE. So there is no auto complete getter and setter in visual code editor. However, I enjoyed to code according to the plan rather than code first think later.

Subtask B JSON

Plan

Objective The objective of this subtask is to convert the objects created in *Subtask A* into JSON objects and show them in the terminal using *System.out.println* method.

Requirements of this subtask is to verify that the list of books is printed out as JSON objects in the terminal(where vagrant is run) when calling the URL http://localhost:9090/api/books.

Strategy Below is the strategy for achieving the requirement:

1. Read about JSON from the link provided in *pingResource* class.
2. Implement *toJSON* method in *GetBooksResource* class which will take a *book* object as parameter and covert it into JSON object and return it as a *String*.
3. Use *toJSON* method inside the *System.out.println* method for showing the JSON objects in the terminal.

Reflection

All the implementation went according to the plan and program run successfully in first attempt. In the beginning I thought that converting Java object to JSON might be quite hard but it was very easy. This time my actual time is less than the estimated time because I realised the editor problem in *Subtask A*. However, I would like to more accurate in further tasks.

Improvements and Strategies

1. I would like to move the *toJSON* method from *GetBooksResource* class to *book* class. The reason for this is, I need to convert Java objects to JSON objects in order to display the them and so far I have only one object which is *book*. Also, it is more logical to place the *toJSON* method in *book* class because having this method in other class is same as if I create *toString* method for *book* class in some other class.
2. I would like to predict my estimate time more accurately because it effects on my project time. Also, in general completing task in limited/given time is very good practice.

Subtask C Web

Plan

Objective The objective of this subtask is to answer the request in the web browser in order to display the JSON object on screen.

Requirements Following are the requirements of this subtask: .

1. Implement the improvemnets.
2. Verify that list of books shows as JSON objects shows on the screen when calling the URL http://localhost:9090/api/books

Strategy Below is the strategy for achieving the requirements:

1. Make estimate time prediction more accurately
2. Move *toJSON* method into *book* class.
3. Look at the *pingcode* for inspiration.
4. Look some online examples on how to create JSON array structure.
5. Update *getBooks* method in *GetBooksResource* class. After creating and adding the books in the list, convert each book into JSON object and save it in a *String*. Do not forget to add ',' to separate the JSON objects. At last return the *String* by adding square brackets in the beginning and at the end.

Reflection

All the implementation went according to the plan and program run successfully in first attempt. I spent some time to see the pattern of separating JSON objects in a string. However, my estimate time is again less than estimate time. It is very difficult to estimate when coding new concepts as compare to old one. So, practice is the only solution to know more about my own abilities and weaknesses.

Task 2 Vision

Reflection

For writing the vision document, I took long time because the explanation of the given project in the *Assignment Overview and Application Stack* is very wide and I should have to narrow it down so that if some new person read this document, it should be easy to understand the system. On the other hand, it should also be easy for the developers or the team members to know what is the system in general ?, what are their tasks? and what is the process for completing their tasks in order to complete the development. It was challenging to narrow down the information because everything is related to each other. However, analysis is always good in order to remove the unnecessary information for making a clear picture of the system.

Task 3 Project Plan

Reflection

It was very challenging for me get the correct structure of the project plan because I have never made it. Somehow, I tried to put all the section which is discussed in the lecture. I wrote the whole project plan on 28 Jan but I was not fully confident about it. After getting suggestions from teaching assistants I rewrite the plan again in more clear and structural form. In the first version, I added strategy plan for completing the milestones and one of the teacher assistant told me that it is not necessary but I can still add it. However, I decided to remove it because it describes the very detail parts of how to implementation in order to complete the milestones.