Student Management System (SMS)

@Author : Shakti Singh Shekhawat

@Publish Date : July 3rd, 2019

@Version : 1.0

Table of Content:

1. Introduction :

This document explains the general design consideration, technology stack, deployment manual and user manual for Student Management System (hereafter referred as SMS).

SMS core function is to provide student portfolio management system which should allow creation (Add), update and view feature for students details. All the interaction with services should happen through user interface.

2. Installation Manual :

The installation manual has been written with the assumption that the machine on which the student management portal needs to run already have Java 8, maven, git and nodeJS pre-installed.

1 . **Source code path :**
   a. **Back end rest api (Spring boot) :** https://github.com/Shakti096/SMS
   b. **Frone end Angular 6 source code :** https://github.com/Shakti096/SMS_FE
   c. **Back end executable :** https://github.com/Shakti096/SMS/blob/master/student-management-app-0.0.1-SNAPSHOT.jar

2 **Spring boot app (Backend api run) :**
Executable jar : Jar can be downloaded from git location and can be started using below command. To run go to the root directory of the SMS project (C:\users\SMS) using command prompt and run the application using below commands :

   i. java –jar student-management-app-0.0.1-SNAPSHOT.jar

3 **Angular app run (SMS_FE) :** Need to download source code from given git repo SMS_FE url. After download need to go to root folder of SMS_FE folder and run below command.

- **Npm install (if not installed)**
- **ng serve**

Now you can start application with browsing localhost below link.

Link : http://localhost:4200/

- Alternatively the project can be build using maven by downloading the source code.

1. Downloaded the source code from Github using below URL in a folder (for eg. : C:\users\SMS).

   $ git clone ……..

   $ git clone ……..

2. Move to the root directory of the SMS project (C:\users\SMS) using command prompt

   $ cd C:\users\SMS

3. Run the below maven command for student-management-services project to build and generate the executables and 4. move to target folder where executable can be found with the name …….

   $ mvn clean install

4. Then start the service using below maven or Java command

   $ mvn spring-boot:run target/…..

5. Then move to the root directory where frontend code has been downloaded and execute below two commands to generate the artifacts and start the UI.

   i. **Npm install (if not installed)**

   ii. **ng serve**

6. Now run the URL in chrome browser to access the portal.

i. http://localhost:4200/

3 . Technology Stack:

Major technology stack used for building SMS have been outlined below along with their advantages and limitations:

1. Spring Boot
2. Angular 6
3. Maven
4. H2 in memory database
5. JPA (spring-data-jpa)
6. RESTful Web Service
7. Apache tomcat server
8. Java 8

1. Spring Boot:

To implement SMS spring boot has been used in order to avoid initial the server set-up, spring configurations and enable rapid development so that time to market can be reduced significantly. Also it can be easily integrated with any of the public cloud like Amazon AWS, Microsoft Azure etc. At the same time in-build database and tomcat server can be used to run the application without the need of installing them.

Advantages:

- Spring Boot makes it easy to create spring-powered, production ready application and services.
- Enables radically faster and widely accessible experience for all Spring development
- Provide a range of non-functional features that are common to large classes of projects (e.g. embedded servers, security, metrics, health checks, externalized configuration)

- Absolutely no code generation and no requirement for XML configuration

Limitations:

- Time consuming process to convert existing or legacy Spring Framework projects into Spring Boot Applications.
- Spring Boot is limited to work with Spring based applications (Java/groovy).

3. **Angular 6**: Angular and html 5 has been used to develop front end for SMS application. This is very advance framework which is used now a days . This provides a lot of advantage in development of front end code like :

   Advantages:

   - Provide a nice structure to develop the code with easy installation using CLI.
   - Code generation can be done using simple cli commands like code for components, services…
   - Complex applications can be developed easily as this provide a fine routing , directives, two way binding.
   - Dependency injection and services features are provided to connect back end api easily.
   - Inbuilt testing framework provided to test front end components.

3. Maven:

Maven has been used as build tool for the application. It is used to manage dependencies of the application. It is essentially a project management and comprehension tool and as such provides a way to help with managing Builds, Dependencies, Documentation, Releases and Distribution.

4. H2 in memory Database:

Typical databases involve a lots of setup like install the database, setup schema, setup the tables, application to database connection. Therefore for the SMS in memory H2 database has been used which is provided by spring boot.

Advantages:

- Zero project setup or infrastructure
- Zero Configuration or maintenance
- Easy to use for Learning, POCs and Unit Tests

Spring Boot provides Simple Configuration to switch between a real database and an in memory database like H2

Limitations:

- Poor query optimizer.
- Concurrency issues.

5. Spring-data JPA:

To implement the data access layer spring-data-jpa has been utilized in the application. It helps in reducing boiler-plate code and reduces the effort and amount needed to maintain data access layer.

Advantages:

- Support to build repositories based on Spring and JPA
- Easy migration to other JPA for data access layer.
- Input validation support for entity and objects.

6. RESTful Web Service:

Rest is an architecture style for a web service to deal with client/server architecture and is designed to use a stateless communication protocol, typically HTTP. Simplicity, lightweight and faster response makes it more likeable option for services.

Advantages:

- Resource identification using URI.
- Uniform interface.
- Self-descriptive messages
- Stateless interactions through hyperlinks

Limitations:

- Only works on top of the HTTP protocol.
- Hard to enforce authorization and security on top of it.
- Hard to enforce a strict contract between client and server.

7. Java 8:

Java 8 is one of the major milestone in the history of Java programming. It has enabled lots of developers across world to reduce boilerplate code and focus more on implementation of business logic. Improved performance, better data structure, memory management has made it developers first choice.

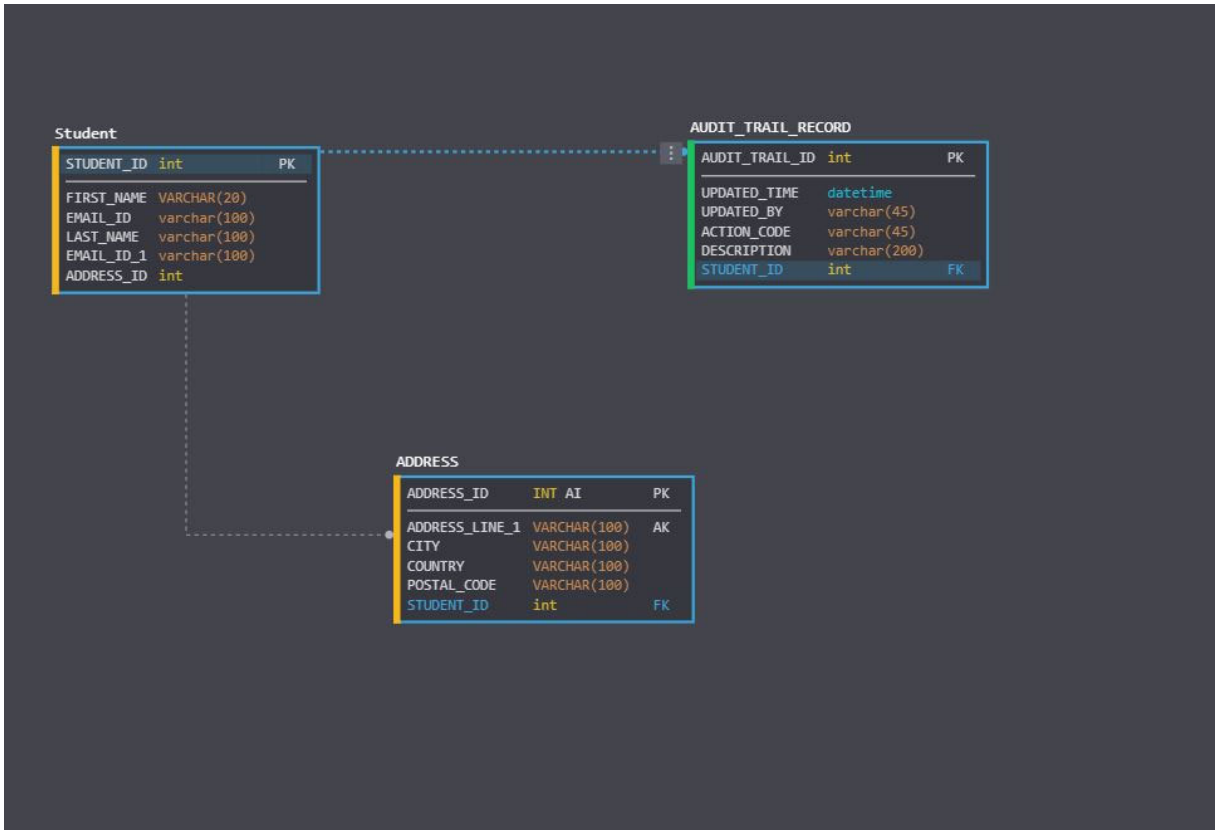
Advantages:

- Improved performance.
- Better memory management.
- Improved productivity.

4 . Data Model:

To implement Student Management services minimum set of relational tables have been used with mandatory fields, which is described below:

1. STUDENT ----------1:M----------AUDIT_TRAIL_RECORD
2. STUDENT ----------1:1----------ADDRESS



5 Rest Services :

Following the REST convention and HTTP protocol specification, the REST APIs are designed:

Si No	Usage	API
1	Create Student	/ student/create
2.	GET All students	/students
3.	GET specific students details	/student/{studentId}
4.	Update students details	/ student/update

6 References :

While building the application and selecting the technology stack below references have been used :

1. https://maven.apache.org
2. https://spring.io/projects/spring-boot
3. https://projects.spring.io/spring-data-jpa/
4. https://spring.io/guides/gs/rest-service/
5. https://www.apache.org/
6. https://angularjs.org/
7.https://www.oracle.com/technetwork/java/javase/overview/java8-2100321.html
8. http://www.springboottutorial.com/spring-boot-and-h2-in-memory-database