

Document name	<i>Java Major Project (Module)</i>
Version no.	<i>1.0</i>
Release date	<i>17th May 2021</i>
Classification	<i>Departmental</i>

This document of Cybage Software Pvt. Ltd. is for restricted circulation. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means – recording, photocopying, electronic and mechanical, without prior written permission of Cybage Software Pvt. Ltd.

Java Major Project (Module)

Document History

Ver. No	Release Date	Created By / Modified By and Date	Reviewed By and Date	Approved By and Date	Remarks and Changes Made
0.1	17 th May 2021	Dadaram Jadhav			Initial draft.

Feedback system

Project Statement:

Develop independent feedback system service

Abstract of the System:

Need to develop independent feedback system service. Feedback system needs to be generic (not coupled to any system). It needs to be developed as independent service and deployed separately. It is useful to collect feedback of any system like experience with doctor's consultation, experience with tutor, experience of purchased product, experience of course attended, etc. Consume functionality of feedback service in your application. Admin will setup feedback form through ready template. Users will fill up feedback form. Only application backend will be managed with the help of REST API. No need to create GUI for application. GUI will be developed by consumer of the application.

System Roles:

- Admin
- Users

Admin Role:

- Setup feedback form using feedback template
- Make feedback visible/invisible as case may be
- View feedback summary

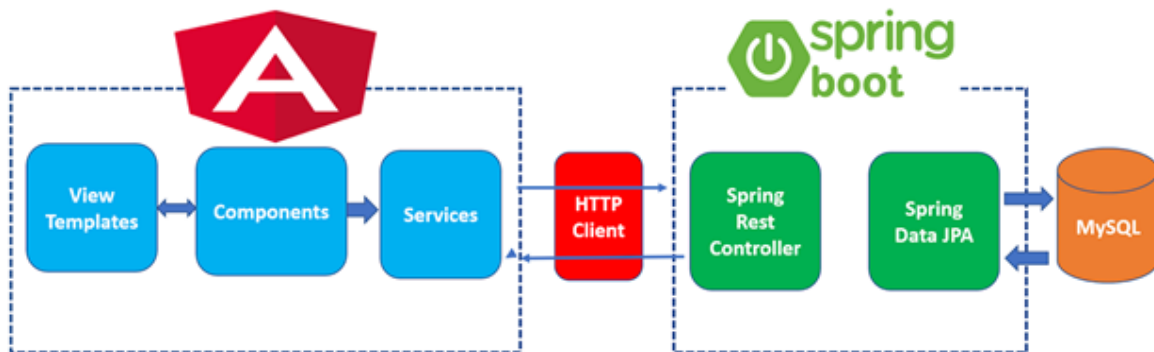
Users Role:

- Feedback form will be visible at the end of transaction. E.g. after completion of doctors consultation, after attending course, after purchasing product.
- Fill up feedback form
- Once filled up feedback form, option will be disabled automatically.

Technologies to be used:

- Spring Architecture
- Spring Rest for Web Services
- Spring Data JPA
- MySQL
- Static code validator – Linter/Hinter
- Editors – Eclipse, STS, etc.
- Code coverage reports – SonarQube
- Git repository (commit after every important milestone)
- Other DevOps tools to be implemented.

System architecture:



Note

- **Make assumptions whenever required in project**
- **Ensure following topics are implemented in project**
 - Classes, object
 - Encapsulation
 - Inheritance
 - Abstraction
 - Polymorphism
 - Static variables/methods
 - Lambda expression
 - Stream API
 - Exception handling (status code, custom message)
 - Architecture: micro services
 - Design pattern: factory, builder design patterns
 - Back end layer: controller, service, repository