|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name : | Ngo Van Phong |  | Date : | 8-3-2014 |

|  |
| --- |
| **General** |

1. What version of Java (JDK) that you have worked with?

|  |  |
| --- | --- |
| JDK Version | Years of Experience |
| 17 | 1,5 |
| 11 | 1,5 |
| 8 | 1,5 |

2. What are some of the IDEs (that you have used) and your choice of IDE/code editor when developing JAVA applications?

IntelliJ, VSCode

I choose IntelliJ to develop Java Applications thanks to its better support and plugins

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| Skill Level   * **N** = Novice level; does not know much * **B** = Beginner level; know how to write/code/use * **I** = Intermediate level; have few years of experience coding/using and is comfortable * **A** = Advance level; expert in using/coding in (almost) all the aspects and knows the nuances |

3. What are some of the revision source control systems that you have used?

|  |  |
| --- | --- |
| Name | Version/Release, Skill Level (N/B/I/A) or Years |
| Git | I |
| SVN | N |
| Mercurial | N |
|  |  |

4. What are some of the Java build tools that you have used?

|  |  |
| --- | --- |
| Name | Version/Release, Skill Level (N/B/I/A) or Years |
| Apache Ant | N |
| Apache Maven | I |
| Gradle | B |
|  |  |

5. What are some of the unit test frameworks/tools that you have used?

|  |  |
| --- | --- |
| Name | Version/Release, Skill Level (N/B/I/A) or Years |
| JUnit | B |
| TestNG | N |
| Selenium | B |
| JMeter | N |
|  |  |

6. What are some of the application frameworks that you have used?

|  |  |
| --- | --- |
| Name (and Version) | Version/Release, Skill Level (N/B/I/A) or Years |
| Hibernate | I |
| Struts | N |
| Spring | I |
|  |  |
|  |  |

7. What are some of the viewing/presentation technologies/frameworks that you have used?

|  |  |
| --- | --- |
| Name (and Version) | Version/Release, Skill Level (N/B/I/A) or Years |
| Angular / Angular.js | N |
| Vue.js | N |
| Bootstrap | I |
| React.js | I |
|  |  |

8. What kind of J2EE application servers that you have used?

|  |  |
| --- | --- |
| Name (and Version) | Version/Release, Skill Level (N/B/I/A) or Years |
| Tomcat | B |
| JBoss AS | N |
| GlassFish | N |
|  |  |

9. What kind of database systems that you have worked with?

|  |  |
| --- | --- |
| Name (and Version) | Version/Release, Skill Level (N/B/I/A) or Years |
| MySQL | B |
| Oracle RDBMS | I |
| MongoDB | I |
|  |  |

10. What kind of CI/CD systems that you have worked with?

|  |  |
| --- | --- |
| Name (and Version) | Version/Release, Skill Level (N/B/I/A) or Years |
| Jenkins | B |
| SonarQube | N |
| Docker | I |
| Kubernetes | N |
|  |  |
|  |  |

11. What kind of Cloud based systems that you have worked with?

|  |  |
| --- | --- |
| Name (and Version) | Version/Release, Skill Level (N/B/I/A) or Years |
| Amazon Web Services (AWS) | N |
| Google Cloud Platform (GCP) | N |
| Microsoft Azure | B |
|  |  |
|  |  |

12. Do you have experience with these technologies?

|  |  |
| --- | --- |
| Name (and Version) | Version/Release, Skill Level (N/B/I/A) or Years |
| Node JS | I |
| TypeScript | I |
| jQuery | B |
|  |  |
|  |  |

1. Coding Tasks

**Guidelines**

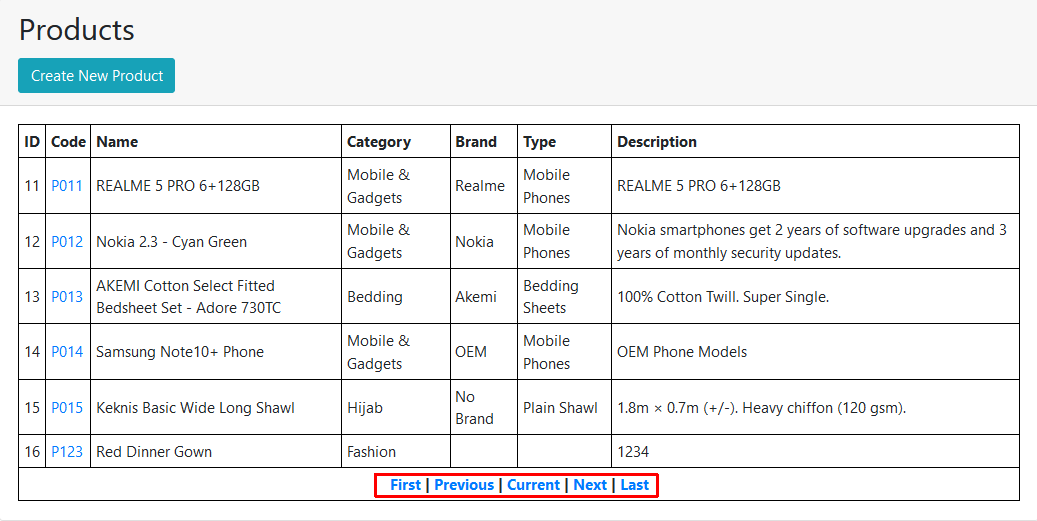
1. Complete as many tasks as possible, within the stipulated time frame (~ 1 mandays).
   * If you are of junior level, complete at least one task.
2. Provide the project source code, minus the binaries (\*.war, \*.jar, \*.ear), that can be built from scratch and run.
3. Grading criteria:
   * Correctness (functioning, bug free, performance)
   * Style (code formatting, naming convention, readability)
   * Comprehensiveness (flexible, reusable, validation, unit tests)
4. Providing the user guide on how to prepare environment, build from source code and run the project. (Better with screenshot of each steps).

**Task 0**

* Build a suitable presentation layout (UI template) for the application.
* UI can be bland, using simple HTML (<div>, <table>) only.
* Or if you are capable/experienced, use Bootstrap (or angular, or react.js, or vue.js) as starting base.
* Screenshots provided served as guideline and minimum requirements. Feel free to express your creativity.

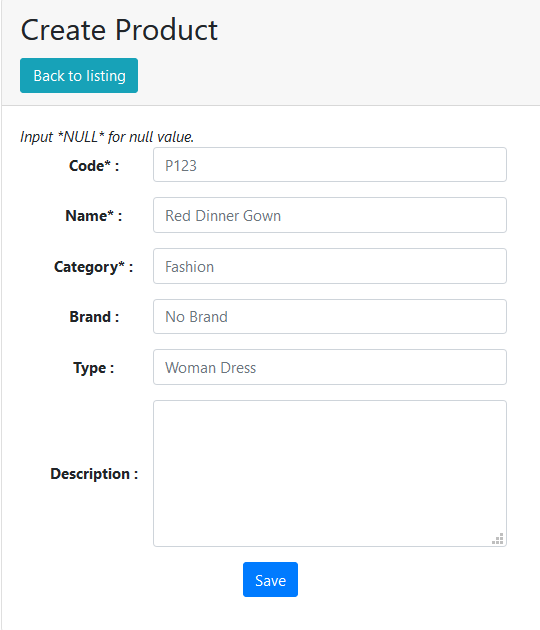
**Task 1**

* Build a HTML page to display the list of the products from the data store, in tabular format.
* Must show (at least) these product information (name, category, description; refer to model definition) as table columns.
* Preferably have pagination (<https://www.google.com/search?q=pagination+listing>)
  + - "page": page number counting starts from 1
    - "size": default of 10 records per page and maximum 25 records per pagecatede



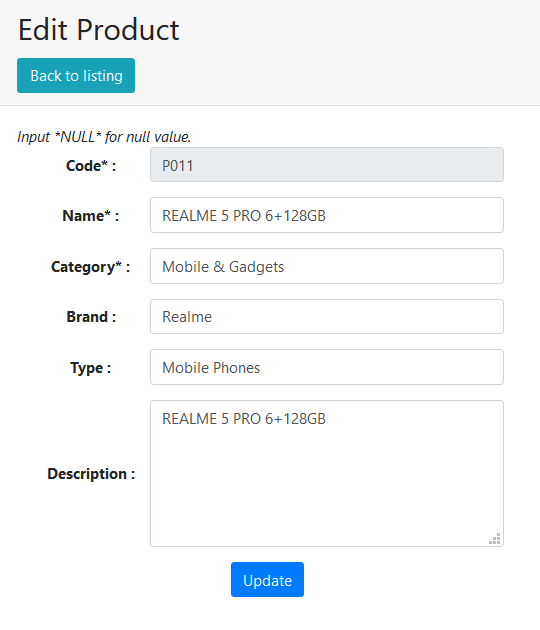
**Task 2**

* Build a "create product" (FORM POST) page/component to add new product item.
* Should have proper validation logic and display error/success messages.
* It can be modal popup dialog style or it can be shown on separate page.
* Implement the corresponding logic at the server side to allow adding new product item to the data store.



**Task 3**

* Build an "edit product" (FORM POST) page/component to update existing product item.
* Should have proper validation logic and display error/success messages.
* It can be modal popup dialog style or it can be shown on separate page.
* Implement the corresponding logic at the server side to allow update of product item from the data store.



**Advance Task**

* Create a REST Controller which provides endpoints for caller to interact with the "product" resource.
* Design and implement the logic, which include:
  + listing of products (with pagination support) => GET /api/products
  + retrieve single product => GET /api/products/{code}
  + create new product => POST /api/products { request body in JSON }
  + update product => PUT /api/products/{code} { request body in JSON }
  + delete product => DELETE /api/products/{code}