**Requirement**

Company X wants to expand their online presence to have a Customer Portal. The following features needs to be enabled.

- As a customer I want to be able to register on the portal with my user account

- As a customer I want to be able to log into my account

 - As a customer I want to update my personal details such as

     - Contact numbers

     - Billing address

- As a customer I want to search products available on the catalogue

- As a customer I want to add selected products into a cart

- As a customer I want to check and pay for my item

Divided SDLC phases based on the above requirement and provided estimations as shown below

* Design
* Coding
* Testing
* Deployment

|  |  |
| --- | --- |
| **Tasks** | **Duration** |
| Design document | 3 weeks |
| Unit test cases using JUNIT for Service, DAO | 1 week |
| Code analysis changes based analysis tool (SONAR,PMD..etc) | 1 week |
| SIT Testing | 2 weeks |
| SIT bug fixes | 2 weeks |
| QA testing | 2 weeks |
| QA bug fixes | 2 weeks |
| UAT testing | 3 weeks |
| UAT bug fixes | 2 weeks |
| Release preparation , deployment plan | 1 week |
| Project meetings | 2 weeks |
| Buffer / Risk | 2 weeks |

**Total**  **23 weeks**

**Development Estimation**

|  |  |  |
| --- | --- | --- |
| **Module** | **Complexity** | **Details** |
| Authentication | High | * Implement UI interface * Create spring security configuration class * Users authentication and authorization * Error handling mechanism * Create user details service class * Create DAO layer * Database configuration * Model object and entity objects * Logger framework functionality across all classes |
| User Registration | Medium | * Implement UI interface * Authenticate Users roles and privileges * Validate user input data * Create user search service * Create DAO layer repository * Model object and entity objects * Database configuration * Logger framework functionality across all classes |
| User Update | Medium | * Implement UI interface * Authenticate Users roles and privileges * Validate user input data * Create user update service * Create DAO repository * Model object and entity objects * Logger framework functionality across all classes |
| Search Product | High | * Implement UI interface (different search patterns like auto completer, multi drop down, date range fields …etc. ) * Authenticate Users roles and privileges * Create user search product service * Create DAO layer repository * Model object and entity objects * Database configuration * Logger framework functionality across all classes |
| Add Product | Medium | * Implement UI interface * Error handling mechanism * Create user search service * Create DAO layer repository * Model object and entity objects * Handling message broker events |
| Payment Gateway | High | * Implement UI interface * Authenticate Users roles and privileges * Error handling mechanism * Create payment service class * Create DAO layer * Database configuration * Model object and entity objects * Handling message broker events |
| Miscellaneous tasks | High | * Version control * Micro service Load balancer * Configuration server * Service fail over scenarios * ELK * Message server and listeners |

**Total Number of hours 800 hours (Approx. 22 weeks)**

**Complexity Table**

|  |  |
| --- | --- |
| **Complexity** | **No of Hours** |
| Small | 20-40 |
| Medium | 40-80 |
| High | 80-140 |

Total estimation for Coding 22 weeks

Total estimation 23 weeks

(Which includes design, testing [SIT/QA/UAT], code rework, bug fixes, release plan, project meetings, Risk]

**Total duration 45 weeks**