**Code Review Checklist**

|  |  |  |  |
| --- | --- | --- | --- |
| Estudiante | Andrés Felipe Mape | Fecha | 02/04/2017 |
| Programa | Programa 5 | Programa # |  |
| Instructor |  | Lenguaje | Java |

|  |  |  |
| --- | --- | --- |
|  |  | **x** |
| Variables | Nombres en camelCase. | x |
| Están definidas en la parte superior (clase o método). | x |
| Los nombres son claros. | x |
| Excepciones | El código tiene bloques de control try-catch en los métodos significativos | x |
| Las excepciones no conocidas se manejan como Exception. | x |
| Métodos | El nombre del método es claro y no tiene ambigüedad. | x |
| El nombre del método está en PascalCase. | x |
| Los nombres de los parámetros son claros. | x |
| Los parámetros están en camelase. | x |
| La estructura interna del método es clara (incluye comentarios). | x |
| Los condicionales utilizados son claros. | x |
| Los ciclos utilizados son claros. | x |
| Se controlan los parámetros inválidos. | x |
| No existen constantes quemadas que no sean CONST. | x |
| Comentarios | Cabecera de clase. | x |
| Métodos de clase. | x |
| Se realizan las referencias de comentarios a la interfaz. | x |
| Clase | El código está alineado y formateado | x |
| Los Import están todos referenciados en la parte superior (solo los necesarios y ordenados). | x |
| Las regiones están bien organizadas y son claras. | x |
| El interlineado entre métodos debe ser un espacio. | x |