**Checklist de Revisión del Diseño**

|  |  |  |  |
| --- | --- | --- | --- |
| Estudiante | José Guillermo Saldaña Cárdenas | Fecha | 8 mar. 20 |
| Programa | ITC11 | # Programa | 3 |
| Instructor | Adriana Gonzalez | Lenguaje | C++ |

|  |  |
| --- | --- |
| Propósito | Guiarte en la conducción de una revisión efectiva de tu diseño |
| General | * Escribe en el primer renglón el nombre de las partes que vas a revisar (aquellas que son nuevas o que van a ser modificadas) * Revisa tu diseño parte por parte. No continúes con la siguiente parte hasta que no hayas revisado por completo la parte anterior. * Cuando estés revisando una parte, revisa una categoría a la vez * Cuando termines de revisar una categoría para una parte, márcala como revisada |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nombre de las partes: | | Main | Main |  |  |
| Comprensión de Requerimientos | Asegurarme de haber leído los requerimientos atentamente para entender todo lo que se pide y lo que no. | ✓ | ✓ |  |  |
| Flujo de procesos | Entender lo que involucra cada requerimiento, identificar que procesos se tendrán que diseñar, sus pasos que se tendrán que implementar, y verificar que no haya algún diseño que facilite este proceso de una manera más sencilla. | ✓ | ✓ |  |  |
| Casos límite | Identificar casos limite en procesos definidos del diseño y del uso principal del programa. Verificar que el proceso inicial los maneje, planear para manejarlos de alguna manera o categorizarlos como casos de excepción. | ✓ | ✓ |  |  |
| Identificación de acciones y clases | Generalizar grupos de acciones para encapsularlos en algún tipo de clase o controlador. | ✓ | ✓ |  |  |
| I/O | Revisar todos los inputs y outputs que habrá, que se tomen en cuenta para facilitar el diseño, y definir los outputs para posibles casos de excepción. |  | ✓ |  |  |
| Cohesión de clases | Que el diseño final realice totalmente lo que se pide en los requerimientos establecidos, y que las clases que lo integran funcionen coherentemente entre sí |  | ✓ |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nombre de las partes: | | Analizador |  |  |  |
| Comprensión de Requerimientos | Asegurarme de haber leído los requerimientos atentamente para entender todo lo que se pide y lo que no. | ✓ |  |  |  |
| Flujo de procesos | Entender lo que involucra cada requerimiento, identificar que procesos se tendrán que diseñar, sus pasos que se tendrán que implementar, y verificar que no haya algún diseño que facilite este proceso de una manera más sencilla. | ✓ |  |  |  |
| Casos límite | Identificar casos limite en procesos definidos del diseño y del uso principal del programa. Verificar que el proceso inicial los maneje, planear para manejarlos de alguna manera o categorizarlos como casos de excepción. | ✓ |  |  |  |
| Identificación de acciones y clases | Generalizar grupos de acciones para encapsularlos en algún tipo de clase o controlador. | ✓ |  |  |  |
| I/O | Revisar todos los inputs y outputs que habrá, que se tomen en cuenta para facilitar el diseño, y definir los outputs para posibles casos de excepción. | ✓ |  |  |  |
| Cohesión de clases | Que el diseño final realice totalmente lo que se pide en los requerimientos establecidos, y que las clases que lo integran funcionen coherentemente entre sí | ✓ |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nombre de las partes: | | Lector |  |  |  |
| Comprensión de Requerimientos | Asegurarme de haber leído los requerimientos atentamente para entender todo lo que se pide y lo que no. | ✓ |  |  |  |
| Flujo de procesos | Entender lo que involucra cada requerimiento, identificar que procesos se tendrán que diseñar, sus pasos que se tendrán que implementar, y verificar que no haya algún diseño que facilite este proceso de una manera más sencilla. | ✓ |  |  |  |
| Casos límite | Identificar casos limite en procesos definidos del diseño y del uso principal del programa. Verificar que el proceso inicial los maneje, planear para manejarlos de alguna manera o categorizarlos como casos de excepción. | ✓ |  |  |  |
| Identificación de acciones y clases | Generalizar grupos de acciones para encapsularlos en algún tipo de clase o controlador. | ✓ |  |  |  |
| I/O | Revisar todos los inputs y outputs que habrá, que se tomen en cuenta para facilitar el diseño, y definir los outputs para posibles casos de excepción. | ✓ |  |  |  |
| Cohesión de clases | Que el diseño final realice totalmente lo que se pide en los requerimientos establecidos, y que las clases que lo integran funcionen coherentemente entre sí | ✓ |  |  |  |