**ANÁLISIS DE REQUERIMIENTOS**

**Nombre del proyecto: Control de Gastos**

**Líder de Proyecto: Jovanny Zepeda**

1. **Objetivos del proyecto: *Satisfacer las necesidades del cliente a través de un sistema desarrollado, siendo este sencillo de utilizar e implementar.***
2. **Definiciones:**
   * **Rubro: Clasificación de Gasto o ingreso presentado que puede tener múltiples derivaciones tales como hogar, auto, escuela, etc.**
   * **Subrubro: tipo de gasto o ingreso perteneciente a un rubro por ejemplo agua, luz, gasolina, comida,etc.**
   * **Cuenta: Entidad a la cual se carga el gasto o ingreso obtenido por el usuario, por ejemplo: banco, préstamo, etc.**
   * **CRUD: vista para editar, modificar y eliminar un registro.**
3. **Supuestos: *N/A***
4. **Roles: *Toda la información con respecto a los roles se encuentra especificado tanto como en el plan de proyecto como en el ciclo de vida.***
5. **Descripción general del proyecto: *Se requiere satisfacer la necesidad de controlar los gastos del cliente a través de una aplicación que sea sencilla de utilizar y manejar en el equipo del cliente.***
6. **WBS *Especificado dentro de Trello.***
7. **Esquemas, diagramas y gráficos de apoyo.**

**Todos los esquemas o diagramas asociados al desarrollo estan dentro de la carpeta diagramas.**

1. **Requerimientos funcionales**

|  |  |
| --- | --- |
| **ID** | ***Requerimiento*** |
| *Casos de Uso: 1,2,3 ubicado en carpeta casos de uso* | |
| **1** | *CRUD de Rubro*   * *Roles participantes: Base de datos, sistema, usuario* * *Pasos de ejecución:*  1. *Click en botón Rubro* 2. *Seleccionar opción*    1. *(Registro de rubro)*        1. *llenar campos requeridos.*       2. *Click en guardar.*       3. *Validar campos.*       4. *Almacenar en base de datos.*       5. *Mostrar mensaje de confirmación o falla.*    2. *(Modificación de Rubro)*       1. *Click sobre rubro a editar.*       2. *Modificar información de rubro.*       3. *Click en botón modificar.*       4. *Validar información*       5. *Mostrar mensaje de confirmación o falla.*    3. *(Eliminar rubro esto elimina subrubros y movimientos ligados a subrubros)*       1. *Click en eliminar*       2. *Mostrar mensaje de alerta*       3. *click en confirmar eliminación o cancelar.*       4. *Alterar base de datos.*       5. *Mostrar mensaje de satisfacción o error.*  * *Diagrama de apoyo:* |
|  | Casos de uso: 4,5,6 |
| **2** | *CRUD de Subrubro*   * *Roles participantes: Base de datos, sistema, usuario* * *Pasos de ejecución:*  1. *Click en botón SubRubro* 2. *Seleccionar opción*    1. *(Registro de Subrubro)*        1. *llenar campos requeridos.*       2. *Seleccionar un Rubro*       3. *Click en guardar.*       4. *Validar campos y existencia de rubro.*       5. *Almacenar en base de datos.*       6. *Mostrar mensaje de confirmación o falla.*    2. *(Modificación de SubRubro)*       1. *Click sobre Subrubro a editar.*       2. *Modificar información de sub rubro.*       3. *Click en botón modificar.*       4. *Validar información y existencia de rubro*       5. *Mostrar mensaje de confirmación o falla.*    3. *(Eliminar subrubro esto elimina los movimientos ligados)*       1. *Click en eliminar*       2. *Mostrar mensaje de alerta*       3. *click en confirmar eliminación o cancelar.*       4. *Alterar base de datos.*       5. *Mostrar mensaje de satisfacción o error.* |
|  | Casos de uso: 7,8,9 |
| 3 | CRUD de Movimiento   * *Roles participantes: Base de datos, sistema, usuario* * *Pasos de ejecución:*   *Se basa en el mismo concepto CRUD sin embargo en su eliminación solo se elimina a el mismo.* |
|  | Caso de uso: 10,11,12 |
| 4 | CRUD de cuenta   * *Roles participantes: Base de datos, sistema, usuario* * *Pasos de ejecución:*   *Se basa en el mismo concepto que el rubro y subrubro con la diferencia de afectarlos, en caso de eliminación solo elimina los movimientos ligados al mismo.* |
|  | Caso de uso: 13 |
| 5 | Consulta de movimientos por rubro mostrando gastos e ingresos   * *Roles participantes: Base de datos, sistema, usuario* * *Pasos de ejecución:*  1. *Seleccionar opción consulta.* 2. *Seleccionar consulta por rubro* 3. *Mostrar calendario para rango de fechas.* 4. *Seleccionar rango.* 5. *Generar consulta* 6. *Mostrar datos obtenidos separando gastos de ingresos.* |
|  | Caso de uso: 13 |
| 6 | Consulta de movimientos por cuenta mostrando gastos e ingresos   * Mecanismo: Se basa en la misma idea para generar una consulta de un Rubro |
|  | Caso de uso: 13 |
| 7 | Consulta de movimientos por subrubro mostrando gastos e ingresos   * Mecanismo: Se basa en la misma idea para generar una consulta de Rubro |
|  | Caso de uso: 13 |
| 8 | Mostrar gráfico de gastos tras generar una consulta   * *Roles participantes: sistema* * *Pasos de ejecución:*  1. *Se leen los datos obtenidos por la previa consulta.* 2. *Se muestran los datos en forma gráfica mostrando ingresos y egresos por separado* |
| 9 | Realizar traspasos entre cuentas   * *Roles participantes: sistema, usuario* * *Pasos de ejecución:*  1. *click en editar* 2. *seleccionar select de área a la que se requiere realizar el cambio.* 3. *Click en aceptar(este cambio se aplica dentro del mismo caso de uso de actualización de un registro).* |

1. **Requerimientos No funcionales**

|  |  |
| --- | --- |
| SISTEMAS OPERATIVOS | |
| *Sistemas operativos sobre los cuales deberá poder ejecutarse dicha aplicación* | |
| **1** | *El sistema deberá ser capaz de ser utilizado en diferentes sistemas operativos computacionales* |

|  |  |
| --- | --- |
| TECNOLOGÍAS | |
| *Tecnologías requeridas para la elaboración del software solicitado* | |
| **1** | *Tecnologías open source* |

|  |  |
| --- | --- |
| INTERFACES DE USUARIO | |
| *Describe las interfaces necesarias para poder dar uso de la aplicación solicitada* | |
| **1** | *Interfaz de registro de los diferentes tipos de datos* |
| 2 | *Interfaz de consulta* |
| 3 | *Interfaz de resultados de consulta incluyendo la gráfica de gastos* |

|  |  |
| --- | --- |
| EFICIENCIA Y DESEMPEÑO | |
| *Se describe el desempeño y eficiencia requerida por parte del sistema para mostrar una experiencia de uso agradable al usuario* | |
| **1** | *Generar procesos en tiempo real con un máximo de 2 a 5 segundos por consulta* |

|  |  |
| --- | --- |
| ESCALABILIDAD | |
| *Describe la escalabilidad del sistema* | |
| **1** | *El sistema deberá ser desarrollado bajo objetos para obtener un control de la misma y poder adaptar a cualquier cambio de forma rapida* |

|  |  |
| --- | --- |
| SEGURIDAD | |
| *Se describe el tipo de seguridad contemplada por parte del sistema* | |
| **1** | *Evitar perdida de información* |
| 2 | *Control de acceso para evitar usuarios no deseados*   * *Roles participantes: sistema, Usuario, base de datos* * *Pasos de ejecución:*  1. *Ingresar usuario y password* 2. *Click en loggin* 3. *Consultar usuario en base de datos* 4. *Redirigir a pagina principal del sistema en caso de satisfacción* 5. *Mostrar mensaje de error en caso de fallo* |

|  |  |
| --- | --- |
| INTEROPERABILIDAD | |
| *Describe el nivel de interoperabilidad requerido para el desarrollo del sistema* | |
| **1** | *Aplicación aislada* |

|  |  |
| --- | --- |
| USABILIDAD | |
| *Describe el tipo de usabilidad que deberá tener el desarrollo implicado y características a tomar en cuenta para alcanzar el nivel definido* | |
| **1** | *No requerir mas de 5 clicks para ejecutar alguna acción* |
| 2 | *Mostrar mensajes de alerta para notificar al usuario sobre acciones realizadas en todo momento* |