FACULTAD
DE INGENIERIA
Universidad de Buenos Aires

Autoencoders para reforzar modelos de prevención de fraude

Autor:
Ariel Salassa

Director:
Franco Arito (Mercado Libre)

Esta planificación fue realizada en el curso de Gestión de proyectos entre el 24 de junio de 2021 y el 19 de agosto de 2021.

Índice

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0.0	Creación del documento	24/06/2021
1.0	Se completa hasta la sección 5 inclusive	08/07/2021

Acta de constitución del proyecto

Buenos Aires, 24 de junio de 2021

Por medio de la presente se acuerda con el Ing. Ariel Salassa que su Trabajo Final de la Carrera de Especialización en Inteligencia Artificial se titulará "Autoencoders para reforzar modelos de prevención de fraude" y consistirá esencialmente en un sistema que aporte información complementaria para el entrenamiento de futuros modelos de prevención de fraude de pagos electrónicos. El Trabajo Final tendrá un presupuesto preliminar estimado de 600 hs de trabajo y recursos económicos brindados por la empresa Mercado Libre, con fecha de inicio 24 de junio de 2021 y fecha de presentación pública 15 de junio de 2022.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg Director posgrado FIUBA	Franco Arito Mercado Libre

Franco Arito
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

Mercado Pago es la plataforma fintech dentro del ecosistema de Mercado Libre (Meli). En esta plataforma, que tiene millones de usuarios activos, se ofrecen distintas soluciones tecnológicas que hace posible pagar y cobrar en forma online. Algunos ejemplos de dichas soluciones son: abono de servicios tales como luz, agua, gas, televisión, electricidad; recarga de teléfono celular; recarga pases de transporte; pago de peajes; pagos y cobros con códigos QR; envío de dinero; etcétera.

Dada la gran cantidad de usuarios y transacciones, y la variedad de éstas, ha sido necesario desarrollar modelos de machine learning que sean capaces de detectar transacciones fraudulentas que perjudican reputacional y económicamente a la empresa. (Figura 1).

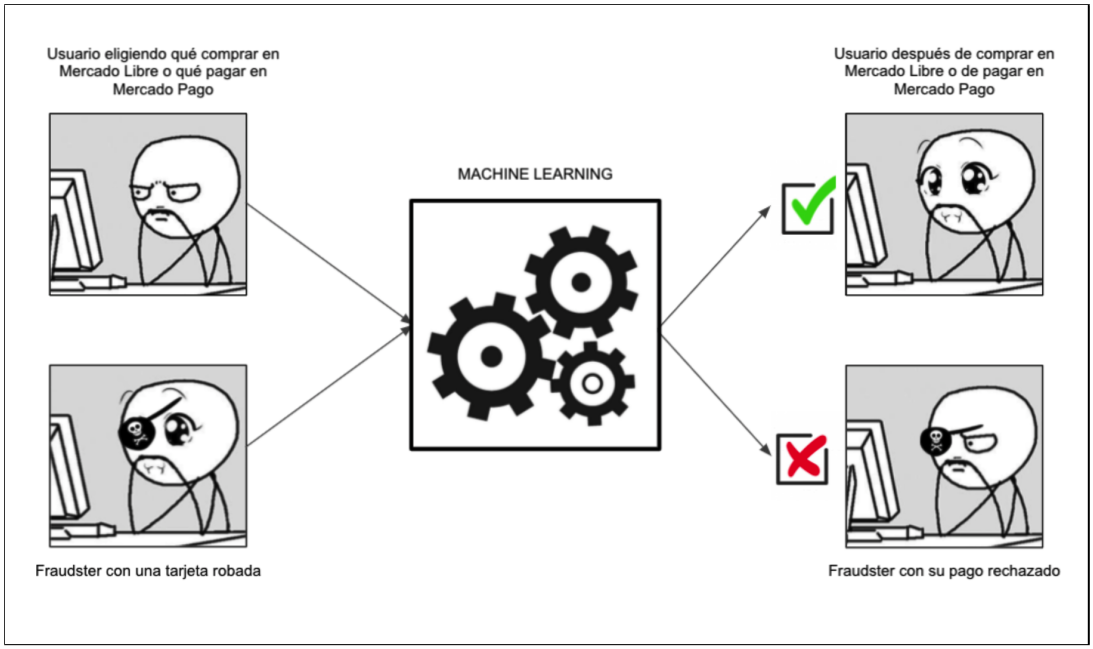

Figura 1. Esquema del comportamiento esperado de los usuarios dentro de las plataformas de Meli.

El entrenamiento de modelos de machine learning para este tipo de aplicaciones conlleva una dificultad adicional: los datos de entrenamiento están fuertemente desbalanceados, es decir, la cantidad de pagos no fraudulentos es mucho mayor que la cantidad de pagos fraudulentos.

Para hacerle frente a este problema, lo que se propone hacer es entrenar un autoencoder (Figura 2) sólo con pagos fraudulentos. De esta manera será posible enriquecer los registros de pagos que en el pasado fueron rechazados por ser riesgos para poder tomarlos en cuenta en futuros entrenamientos. Usar un autoencoder con este tipo de entrenamiento y con este propósito, es decir, entrenado sólo con ejemplos de fraude con el fin de reforzar ejemplos de potenciales pagos fraudulentos para reutilizar la información, es algo que no se ha intentado en la comunidad.

Por otro lado el autoencoder presenta en su capa central o capa latente una representación reducida y codificada de la entrada. Es de esperarse que a partir de ella puedan visualizarse patrones de fraudes conocidos y, eventualmente, sacar conclusiones o indicios de patrones de fraudes sin conocer (Figura 3).

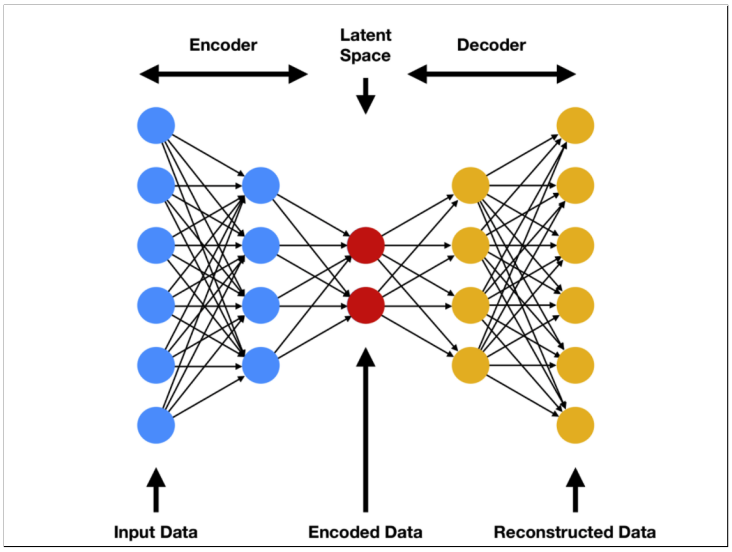

Figura 2. Arquitectura representativa de un autoencoder.

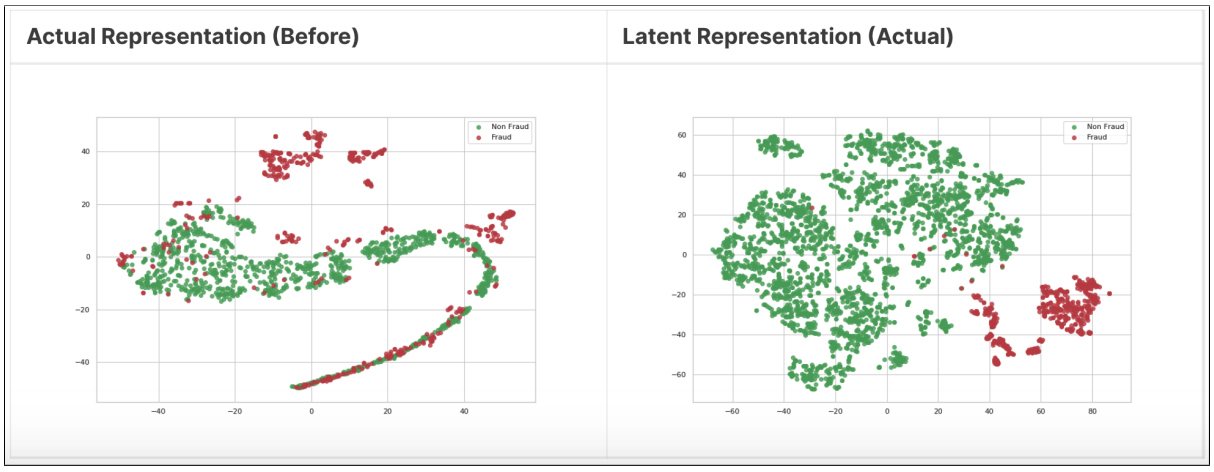

Figura 3. Imagen ilustrativa del artículo *'Semi Supervised Classification using AutoEnconders'* de Kaggle. A la izquierda se visualiza la naturaleza de las transacciones fraudulentas y no fraudulentas usando el método de descomposición T-SNE (*t-Distributed Stochastic Neighbor Embedding*) aplicado a los datos de entrada. A la derecha se observan los mismos pagos representados a partir de la capa latente. En el caso de este trabajo lo que se espera es que la dispersión de los puntos permita clusterizar los ejemplos en distintos perfiles de fraude

En la Figura 4 se observa un diagrama de bloques que ilustra cómo sería el funcionamiento del sistema en producción. En primer lugar, un usuario realizaría un pago en línea. Inmediatamente, el pago entra al sistema y se obtiene una representación vectorizada del mismo con los atributos de interés. Esta codificación del pago pasa por la red neuronal del motor de fraude y se obtiene una probabilidad de que el pago sea fraudulento (predicción). En función de la probabilidad de fraude y otros factores se decide si el pago se rechaza o se aprueba y se guardan todos los valores en una base de datos. En caso de que el pago sea rechazado, el mismo se envía al autoencoder de Fraude. Del autoencoder se obtendrá una puntuación de fraude asociada al pago rechazado que también se guardará en la base de datos. La puntuación de fraude será una medida de la capacidad de reconstrucción del autoencoder y representará qué tan similar a un fraude real es el pago rechazado. Al momento de entrenar nuevos modelos de red o reentrenar modelos existentes, los pagos con alta puntuación podrán ser considerados en el dataset de entrenamiento.

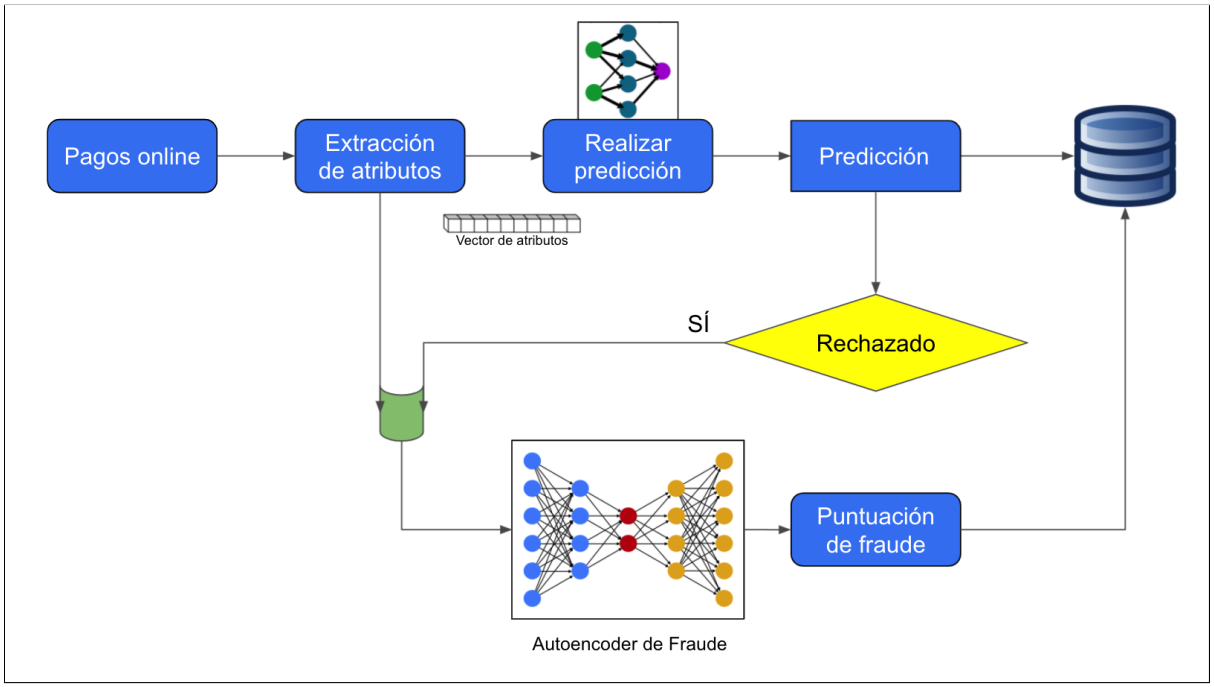

Figura 4. Diagrama de bloques del funcionamiento del sistema

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Responsable	Ariel Salassa	Mercado Libre	ML Engineer Alumno
Colaboradores	Paz Martin Joaquín Loyola Enrique Serdio	Mercado Libre	Sr. Data Scientist Sr. Data Engineer Sr. ML Engineer
Orientador	Franco Arito	Mercado Libre	Sr. ML Expert Director Trabajo final
Usuario final	Desarrolladores de Machine Learning	Mercado Libre	Data Scientists ML Engineers

Cuadro 1. Identificación de los interesados

- Responsable: Ariel Salassa, es la persona que desarrollará el proyecto.
- Colaboradores:
  - Paz Martín: es líder y referente técnica del equipo de científicos de datos donde se desempeña el responsable. Validará la gestión del tiempo y será capaz de orientar en el desarrollo si el responsable lo requiriese.
  - Joaquín Loyola: es líder y referente técnico del equipo de ingeniería de datos. Su colaboración pasará por asistir al referente en cuestiones ligadas a los datos de entrenamiento, si fuese necesario.
  - Enrique Serdio: es referente técnico del equipo de ML Ops. Su colaboración se centrará, si fuese necesario, en asistir al responsable en cuestiones ligadas a la infraestructura de los modelos de machine learning en la nube.

- Orientador: Franco Arito es el director del presente proyecto y líder técnico de múltiples equipos de Mercado Libre. Su función será orientar al responsable a lo largo de la realización del proyecto.
- Desarrolladores de Machine Learning: son los usuarios finales que podrán hacer uso del sistema para enriquecer sus modelos.

3. Propósito del proyecto

El propósito de este proyecto es poner en valor los pagos que son rechazados por el motor de fraude y que tienen potencial de ser utilizados en futuros entrenamientos de redes neuronales de manera tal de reducir el desbalance de los datasets de entrenamiento y validación. Además, se espera que la representación en la capa latente permita evaluar oportunidades para determinar perfiles de fraude. Con una representación como ésta, los equipos de prevención tendrán a su disposición una herramienta que les permitirá ser más reactivos ante posibles ataques.

4. Alcance del proyecto

El proyecto comprenderá las siguientes etapas:

- Planificación de tareas.
- Formación en TensorFlow.
- Investigación de autoencoders aplicados a la prevención de fraude.
- Selección y extracción del dataset para realizar prueba de concepto del modelo.
- Análisis de datos del dataset.
- Pruebas de arquitectura de red.
- Visualización y análisis de datos de la capa latente utilizando el método de descomposición T-SNE.
- Evaluación de distintas formas de hacer etiquetado (labeling).
- Evaluación de la performance del sistema comparado con otras soluciones.
- Evaluación del modelo con otros datasets.

El presente proyecto no incluye:

- Aplicación de algoritmos de clustering para los datos codificados a partir de la capa latente.
- Despliegue del modelo y puesta en producción.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- El responsable dispondrá de suficiente cantidad de tiempo para encarar los problemas que se presenten en el desarrollo del proyecto.
- El responsable tendrá a su disposición a su director y/o colaboradores cuando sea pertinente.
- TensorFlow es el framework de cálculo numérico que dispone de todas las herramientas necesarias para encarar este proyecto.
- El autoencoder entrenado solamente con pagos fraudulentos tendrá buen ratio de reconstrucción de datos a la hora de evaluar pagos rechazados por alto riesgo.
- La puntuación de Fraude (asociada con la medida de reconstrucción de un pago) será un dato de tipo flotante, o bien, un dato de tipo categórico basado en ciertos valores de corte (thresholds).
- Es posible aplicar el método de descomposición T-SNE a los datos codificados y, a partir de su representación en dos o tres dimensiones, se podrán realizar nuevos análisis, por ejemplo, la identificación de clusters de fraudes.
- Una vez que el autoencoder esté entrenado y validado con un set de pagos, su aplicación podría generalizarse.
- El comportamiento de los usuarios que provocan el fraude no mutará mientras tiene lugar el desarrollo de este proyecto.

6. Requerimientos

Los requerimientos deben numerarse y de ser posible estar agruparlos por afinidad, por ejemplo:

1. Requerimientos funcionales
   1.1. El sistema debe...
   1.2. Tal componente debe...
   1.3. El usuario debe poder...
2. Requerimientos de documentación
   2.1. Requerimiento 1
   2.2. Requerimiento 2 (prioridad menor)
3. Requerimiento de testing...
4. Requerimientos de la interfaz...
5. Requerimientos interoperabilidad...
6. etc...

Leyendo los requerimientos se debe poder interpretar cómo será el proyecto y su funcionalidad.

Indicar claramente cuál es la prioridad entre los distintos requerimientos y si hay requerimientos opcionales.

No olvidarse de que los requerimientos incluyen a las regulaciones y normas vigentes!!!

Y al escribirlos seguir las siguientes reglas:

- Ser breve y conciso (nadie lee cosas largas).
- Ser específico: no dejar lugar a confusiones.
- Expresar los requerimientos en términos que sean cuantificables y medibles.

7. Historias de usuarios (*Product backlog*)

Descripción: En esta sección se deben incluir las historias de usuarios y su ponderación (*history points*). Recordar que las historias de usuarios son descripciones cortas y simples de una característica contada desde la perspectiva de la persona que desea la nueva capacidad, generalmente un usuario o cliente del sistema. La ponderación es un número entero que representa el tamaño de la historia comparada con otras historias de similar tipo.

El formato propuesto es: çomo [rol] quiero [tal cosa] para [tal otra cosa]."

Se debe indicar explícitamente el criterio para calcular los *story points* de cada historia

8. Entregables principales del proyecto

Los entregables del proyecto son (ejemplo):

- Manual de uso
- Diagrama de circuitos esquemáticos
- Código fuente del firmware
- Diagrama de instalación
- Informe final
- etc...

9. Desglose del trabajo en tareas

El WBS debe tener relación directa o indirecta con los requerimientos. Son todas las actividades que se harán en el proyecto para dar cumplimiento a los requerimientos. Se recomienda mostrar el WBS mediante una lista indexada:

1. Grupo de tareas 1
    1.1. Tarea 1 (tantas hs)
    1.2. Tarea 2 (tantas hs)
    1.3. Tarea 3 (tantas hs)
2. Grupo de tareas 2
    2.1. Tarea 1 (tantas hs)
    2.2. Tarea 2 (tantas hs)
    2.3. Tarea 3 (tantas hs)
3. Grupo de tareas 3
    3.1. Tarea 1 (tantas hs)
    3.2. Tarea 2 (tantas hs)
    3.3. Tarea 3 (tantas hs)
    3.4. Tarea 4 (tantas hs)
    3.5. Tarea 5 (tantas hs)

Cantidad total de horas: (tantas hs)

Se recomienda que no haya ninguna tarea que lleve más de 40 hs.

10. Diagrama de Activity On Node

Armar el AoN a partir del WBS definido en la etapa anterior.

Indicar claramente en qué unidades están expresados los tiempos. De ser necesario indicar los caminos semicríticos y analizar sus tiempos mediante un cuadro. Es recomendable usar colores y un cuadro indicativo describiendo qué representa cada color, como se muestra en el siguiente ejemplo:

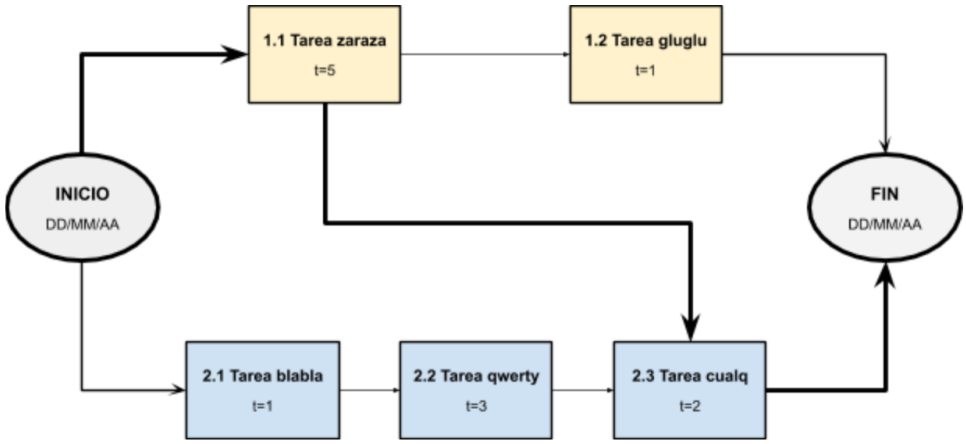

Figura 5. Diagrama en *Activity on Node*

11. Diagrama de Gantt

Existen muchos programas y recursos *online* para hacer diagramas de gantt, entre los cuales destacamos:

- Planner
- GanttProject
- Trello + *plugins*. En el siguiente link hay un tutorial oficial:
  `https://blog.trello.com/es/diagrama-de-gantt-de-un-proyecto`
- Creately, herramienta online colaborativa.
  `https://creately.com/diagram/example/ieb3p3ml/LaTeX`
- Se puede hacer en latex con el paquete *pgfgantt*
  `http://ctan.dcc.uchile.cl/graphics/pgf/contrib/pgfgantt/pgfgantt.pdf`

Pegar acá una captura de pantalla del diagrama de Gantt, cuidando que la letra sea suficientemente grande como para ser legible. Si el diagrama queda demasiado ancho, se puede pegar primero la "tabla" del Gantt y luego pegar la parte del diagrama de barras del diagrama de Gantt.

Configurar el software para que en la parte de la tabla muestre los códigos del EDT (WBS).
Configurar el software para que al lado de cada barra muestre el nombre de cada tarea.
Revisar que la fecha de finalización coincida con lo indicado en el Acta Constitutiva.

En la figura 6, se muestra un ejemplo de diagrama de gantt realizado con el paquete de *pgfgantt*. En la plantilla pueden ver el código que lo genera y usarlo de base para construir el propio.

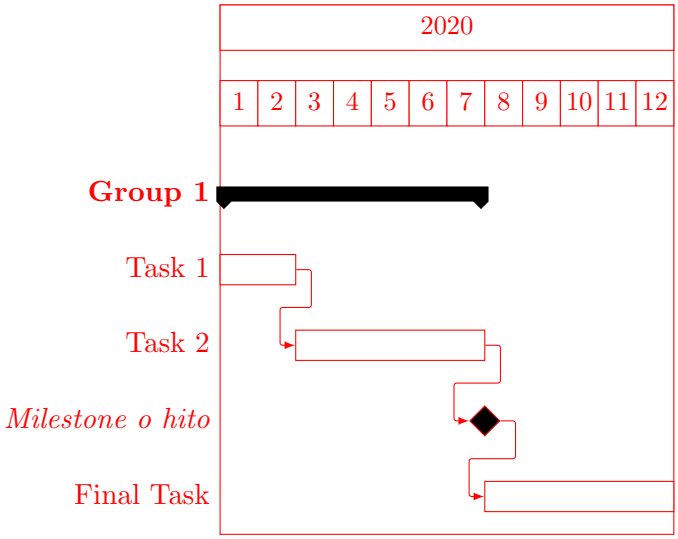

Figura 6. Diagrama de gantt de ejemplo

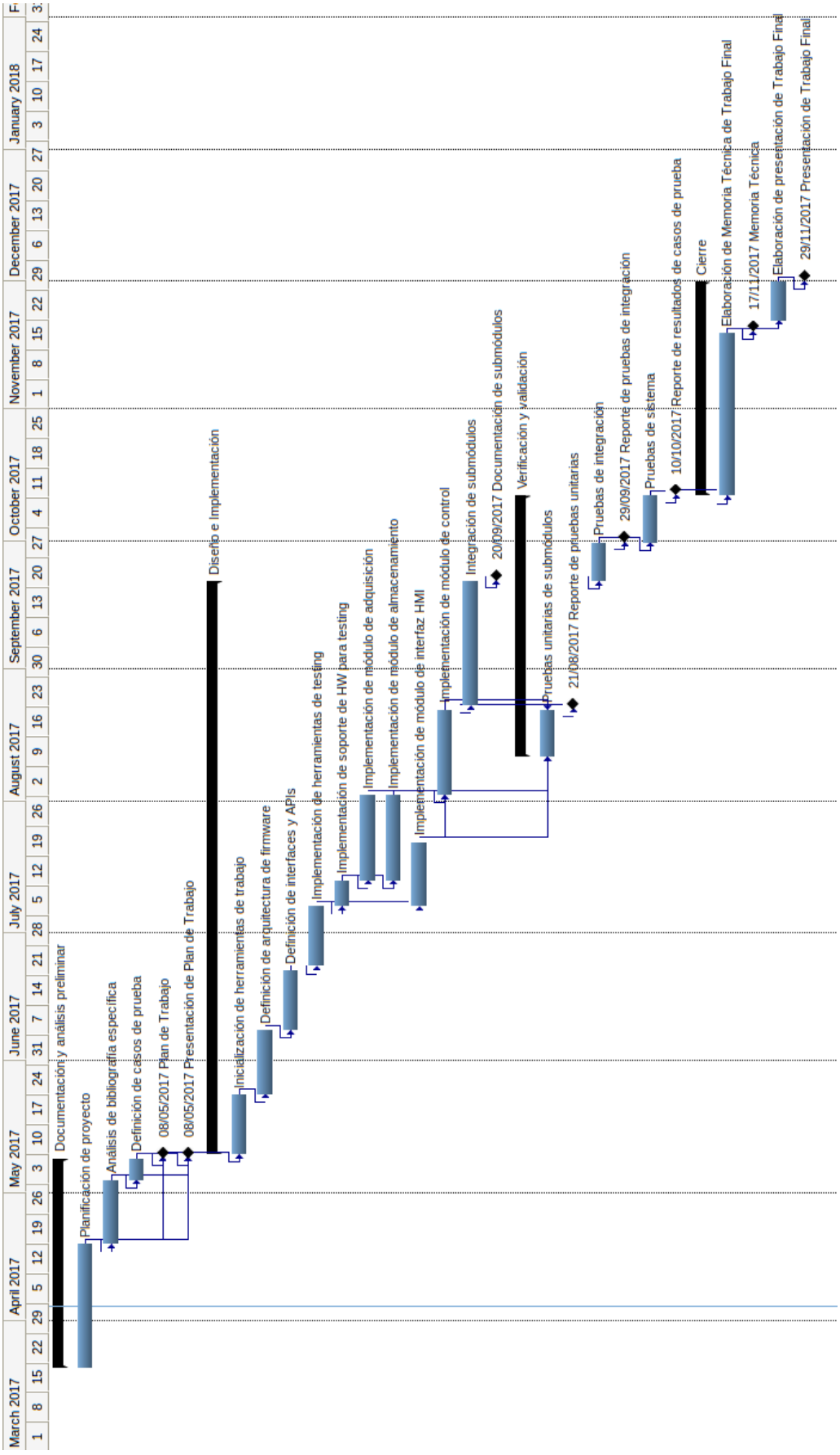

Figura 7. Ejemplo de diagrama de Gantt rotado

12. Presupuesto detallado del proyecto

Si el proyecto es complejo entonces separarlo en partes:

- Un total global, indicando el subtotal acumulado por cada una de las áreas.
- El desglose detallado del subtotal de cada una de las áreas.

IMPORTANTE: No olvidarse de considerar los COSTOS INDIRECTOS.

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
SUBTOTAL			
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
SUBTOTAL			
TOTAL			

13. Gestión de riesgos

a) Identificación de los riesgos (al menos cinco) y estimación de sus consecuencias:

Riesgo 1: detallar el riesgo (riesgo es algo que si ocurre altera los planes previstos de forma negativa)

- Severidad (S): mientras más severo, más alto es el número (usar números del 1 al 10).
  Justificar el motivo por el cual se asigna determinado número de severidad (S).
- Probabilidad de ocurrencia (O): mientras más probable, más alto es el número (usar del 1 al 10).
  Justificar el motivo por el cual se asigna determinado número de (O).

Riesgo 2:

- Severidad (S):
- Ocurrencia (O):

Riesgo 3:

- Severidad (S):

- Ocurrencia (O):

b) Tabla de gestión de riesgos: (El RPN se calcula como RPN=SxO)

Riesgo	S	O	RPN	S*	O*	RPN*

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a...

Nota: los valores marcados con (*) en la tabla corresponden luego de haber aplicado la mitigación.

c) Plan de mitigación de los riesgos que originalmente excedían el RPN máximo establecido:

Riesgo 1: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación). Nueva asignación de S y O, con su respectiva justificación: - Severidad (S): mientras más severo, más alto es el número (usar números del 1 al 10). Justificar el motivo por el cual se asigna determinado número de severidad (S). - Probabilidad de ocurrencia (O): mientras más probable, más alto es el número (usar del 1 al 10). Justificar el motivo por el cual se asigna determinado número de (O).

Riesgo 2: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación).

Riesgo 3: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación).

14. Gestión de la calidad

Para cada uno de los requerimientos del proyecto indique:

- Req #1: copiar acá el requerimiento.
  - Verificación para confirmar si se cumplió con lo requerido antes de mostrar el sistema al cliente. Detallar
  - Validación con el cliente para confirmar que está de acuerdo en que se cumplió con lo requerido. Detallar

Tener en cuenta que en este contexto se pueden mencionar simulaciones, cálculos, revisión de hojas de datos, consulta con expertos, mediciones, etc. Las acciones de verificación suelen considerar al entregable como “caja blanca”, es decir se conoce en profundidad su funcionamiento interno. En cambio, las acciones de validación suelen considerar al entregable como “caja negra”, es decir, que no se conocen los detalles de su funcionamiento interno.

15. Procesos de cierre

Establecer las pautas de trabajo para realizar una reunión final de evaluación del proyecto, tal que contemple las siguientes actividades:

- Pautas de trabajo que se seguirán para analizar si se respetó el Plan de Proyecto original: - Indicar quién se ocupará de hacer esto y cuál será el procedimiento a aplicar.
- Identificación de las técnicas y procedimientos útiles e inútiles que se emplearon, y los problemas que surgieron y cómo se solucionaron: - Indicar quién se ocupará de hacer esto y cuál será el procedimiento para dejar registro.
- Indicar quién organizará el acto de agradecimiento a todos los interesados, y en especial al equipo de trabajo y colaboradores: - Indicar esto y quién financiará los gastos correspondientes.