

**Universidad Tecnológica Nacional  
Facultad Regional Córdoba  
Ingeniería en Sistema de información**



**INFORME TÉCNICO  
Ingeniería de software**

**Ejemplos de grandes proyectos de software  
fallidos y exitosos.**

**Lopez Roberto Martin  
Lopez Ruiz Lautaro  
Murua Ayosa Mauricio  
Reyes Luciano  
Strasorier Ariel**

**Tutora: Covaro, Laura Ines**

**Fecha de presentación: 12/04/2019**

---

## **RESUMEN**

En este informe se abordarán algunos conceptos claves que nos ayudarán a comprender un poco más porque un proyecto de software puede llegar al éxito o al fracaso.

Se explicaran algunas de las condiciones por las que se puede llegar a algunas de estas condiciones, se verán algunos ejemplos de la vida real en los cuales se presentaron estas condiciones, y se realizará un análisis de los casos y se verá cuáles fueron algunos de los puntos claves para el éxito o fracaso de estos proyectos.

---

# **TABLA DE CONTENIDOS**

<b>RESUMEN</b>	<b>2</b>
<b>TABLA DE CONTENIDOS</b>	<b>2</b>
<b>INTRODUCCIÓN</b>	<b>3</b>
<b>1. ¿Que es un proyecto de software?</b>	<b>4</b>
<b>2. Éxito de un proyecto</b>	<b>5</b>
<b>3. Fracaso de un proyecto</b>	<b>6</b>
<b>4. Ejemplos de proyectos fracasados</b>	<b>7</b>
<b>4.1. Sistema automático de manipulación de equipaje del Aeropuerto Internacional de Denver</b>	<b>7</b>
<b>4.2. FBI</b>	<b>8</b>
<b>4.3. El proyecto Stretch de IBM</b>	<b>9</b>
<b>5. Ejemplos de proyectos exitosos</b>	<b>11</b>
<b>5.1. Facturación de Energía Eléctrica</b>	<b>11</b>
<b>CONCLUSIÓN</b>	<b>14</b>
<b>BIBLIOGRAFÍA</b>	<b>15</b>

---

## INTRODUCCIÓN

El éxito en un proyecto de software se debe a que previamente se define el objetivo y visión del proyecto en el negocio, se necesita una capacitación continua del personal que estará a cargo del sistema, por otra parte se necesita contar con asesoría de expertos para la resolución de cualquier problema y dar seguimiento a todos los procesos.

También la organización debe estar abierta al cambio, su sistema debe ser flexible, para ser modificado cuando factores externos o internos afecten su desarrollo.

Por otra parte, muchas empresas han tenido dificultades en los proyectos de desarrollo de software, éstas se deben principalmente a que los costos fueron mayores de los esperados, el tiempo de implementación fue más largo de lo esperado, el sistema de capacitación anterior a la implementación fue deficiente, falta de apoyo por parte de la gerencia, el programa no superó a los objetivos especificados.

Las empresas no deben tenerle miedo al cambio, ni a la aplicación de tecnología de información dentro de su empresa. Es verdad que la implementación de un sistema tiene un costo elevado, que no se sabe cómo funcionará dentro de la empresa, que se necesita el apoyo de la organización; por eso es importante que las empresas analicen y se encuentren bien informadas de todos los cambios que pueda sufrir su empresa, sus beneficios superan a la inversión del sistema.

En diversas organizaciones, se está presentando una situación alarmante con respecto a que la mayor parte de los proyectos informáticos no llegan a buen término, ya sea por que no cumplen con los requerimientos planteados o simplemente son cancelados debido a que superan las expectativas de tiempo y/o costo.

---

# DESARROLLO

## 1. ¿Que es un proyecto de software?

Según la definición de proyecto de software, un proyecto es un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único, que en nuestro caso el producto será el software.

Tienen diversas características:

- Están orientados a objetivos, es decir están dirigidos a obtener resultados y ello se refleja a través de objetivos. Estos objetivos guían al proyecto, no deben ser ambiguos, deben ser claros, alcanzables y medibles (a fin de poder determinar el avance sobre el proyecto).

Los objetivos me dicen dónde empieza y dónde termina el proyecto.

- Tienen una duración limitada en el tiempo, tienen principio y fin, esto implica que cuando se alcanzan el/los objetivo/s, el proyecto termina. El ejemplo más claro de lo que no es un proyecto, es una línea de producción dado que nunca finaliza.
- Implican tareas interrelacionadas basadas en esfuerzos y recursos. Esto se debe a la complejidad sistémica de los problemas. Trabajo en equipo, calendarización, control, monitoreo.
- Son únicos, todos los proyectos por similares que sean tienen características que los hacen únicos. Por ejemplo, tienen cronogramas diferentes.

## 2. Éxito de un proyecto

Un proyecto es exitoso cuando se cumplen los siguientes puntos mas importantes :

- Monitoreo y Feedback: Realizar verificaciones constantes para comprobar que la implementación avanza como se planificó y feedback de tu plan a tu equipo, cliente, etc. Así te aseguras de que es viable y que todo el mundo está de acuerdo.
- Tener una misión/objetivo claro : Si bien muchas veces se da por sentado que todo el mundo entiende las razones de por qué se está llevando a cabo un determinado proyecto,

---

es necesario ser claro y explícito al declarar cuál es el objetivo general del proyecto, y qué objetivos específicos se han determinado que darán por resultado entonces este objetivo general.

- **Comunicación:** La forma en que se comunique al personal lo que se va a realizar es fundamental, porque habitualmente los cambios generan dudas, temor y ansiedad. Por otro lado, nunca hay un exceso de información. Lo ideal es que haya informes frecuentes sobre la marcha del proyecto, y que se habiliten varios canales para hacer llegar las dudas

En el estudio titulado “The CHAOS Report” publicado por Standish Group describe que cuando nos va bien en un proyecto principalmente es por:

1. Involucramiento del usuario 15.9 %
2. Apoyo de la Gerencia 13.0 %
3. Enunciado claro de los requerimientos 9.6 %
4. Planeamiento adecuado 8.2 %
5. Expectativas realistas 7.7 %
6. Hitos intermedios 7.7 %
7. Personas involucradas competentes 7.2 %

### **3. Fracaso de un proyecto**

Los proyectos también pueden fracasar obviamente, ya sea por cualquiera de las siguientes causas:

- Fallas al definir el problema
- Planificar basado en datos insuficientes
- La planificación la hizo el grupo de planificaciones
- No hay seguimiento del plan de proyecto
- Plan de proyecto pobre en detalles
- Planificación de recursos inadecuada
- Las estimaciones se basaron en “supuestos” sin consultar datos históricos
- Nadie estaba a cargo

---

Como ya habíamos indicado anteriormente en el estudio titulado “The CHAOS Report” publicado por Standish Group describe que cuando nos va mal principalmente es por:

1. Requerimientos incompletos 13.1 %
2. Falta de involucramiento del usuario 12.4 %
3. Falta de recursos 10.6 %
4. Expectativas poco realistas 9.3 %
5. Falta de apoyo de la Gerencia 8.7 %
6. Requerimientos cambiantes 8.1 %

A finales del año 2001, Daniel Piorum realizó un estudio con aproximadamente 50 responsables de proyectos, con el objeto de analizar las causas que alimentan los fracasos. En dicho estudio encontró tres principales causas que afectan los proyectos de forma negativa, estas son:

- 21 % Cambios en los objetivos definidos a nivel estratégico
- 31 % No utilización, o mala utilización de metodologías de trabajo
- 48 % Problemas humanos, de conducción, comunicación y conflictos entre las personas.

## **4. Ejemplos de proyectos fracasados**

### **4.1. Sistema automático de manipulación de equipaje del Aeropuerto Internacional de Denver**

El Aeropuerto Internacional de Denver, conocido como DIA por sus siglas en inglés, el mayor en extensión de los Estados Unidos y el sexto en tráfico de pasajeros.

En 1991 el aeropuerto hizo un intento por remodelar y actualizar su penoso y lento sistema de facturación y traslado de equipaje. La idea consistía en colocar etiquetas de códigos de barras en cada maleta para que fueran transferidas en transportadores con carretillas automáticas (DVC, por sus siglas en inglés). Este sistema automatizaría el traslado de equipaje, integraría

---

a las tres terminales y reduciría de forma notable el tiempo de asistencia en tierra de los aviones.

El proyecto de software llamado Sistema Automático de Gestión de Equipajes (Automatic Baggage Handling System -ABHS) fue un fracaso rotundo ya que los objetivos planteados no fueron alcanzados a tiempo ni forma. Sin este software, no se podía atender a los pasajeros ni un solo día.

Basándonos en lo estudiado en clases sabemos que las tres variables principales (the triple constraint) a las que se enfrentan todos los gestores de proyectos son: alcance, tiempo, y costo y el balance de estos tres factores afecta directamente a la calidad del proyecto. Si tuviéramos que evaluar cada una de ellas identificamos de inmediato que el Aeropuerto Internacional de Denver (DIA) no cumplirían con ninguna.

Cuando el aeropuerto encargó a la empresa BAE Systems el desarrollo del sistema automatizado de manipulación de equipaje, pasaron completamente por alto los calendarios de entrega de BAE y se aferraron tozudamente a su irreal previsión de 2 años. El proyecto no estuvo a la altura de lo esperado y la dirección asumió una cantidad innecesaria de riesgos. La decisión que más les perjudicó fue la de no incluir a las aerolíneas en las conversaciones de planificación. Al no tenerlos en cuenta no se diseñaron adecuadamente los medios adecuados con los que manipular el equipaje de grandes dimensiones, soportes para equipamiento deportivo/de esquí y pistas de mantenimiento separadas. En conclusión hubo que rehacer gran parte de trabajos «completados», la apertura del aeropuerto se demoró 16 meses y se incurrieron en pérdidas de aproximadamente 2.000 millones de dólares. El proyecto en su conjunto fue descartado en 2005.

#### Análisis del caso:

Estudiando el caso previamente descrito podemos apreciar que claramente el mismo se ve afectado por algunas de las causas comunes para proyectos fallidos las cuales son “planificar con datos insuficientes” ya que se desarrolló y planificó el proyecto sin tener en cuenta información crucial del negocio, a lo que también afecta a la causa de “falta de involucramiento del usuario” por no incluir a las aerolíneas en las conversaciones lo cual fue su peor error en el proyecto.



---

Podemos ubicar al caso dentro del 13,1% de los proyectos que fracasan por requerimientos Incompletos y 12,4% por falta de Involucramiento de Usuarios.

## **4.2. FBI**

Expediente virtual (FVC ) era un software de aplicación desarrollado por la United States Federal Bureau of Investigation (FBI) entre 2000 y 2005. El proyecto fue abandonado oficialmente en enero de 2005, cuando aún estaba en fase de desarrollo y el costo del gobierno federal cerca de \$ 170 millones.

En septiembre de 2000, el FBI anunció la "trilogía" el programa destinado a la modernización de la oficina de la obsoleta tecnología de la información (TI).

El proyecto constaba de tres partes:

1. La compra de escritorio modernos equipos para todas las oficinas del FBI.
2. El desarrollo de seguridad de alto rendimiento de la WAN y la LAN de redes.
3. La modernización de la suite del FBI de aplicaciones de software de investigación.

Los dos primeros objetivos de la trilogía fueron generalmente exitosos , a pesar de los costos excesivos , pero reemplazar el sistema de software de soporte de casos automatizado (ACS) de Bureau resultó ser bastante difícil. Fue desarrollado internamente por la oficina y se utilizó para gestionar todos los documentos relacionados con los casos investigados por el FBI, lo que permite a los agentes buscar y analizar pruebas entre diferentes casos. El proyecto fue originalmente estaba programado para durar tres años y costaría 380 millones dólares EE.UU. ACS se consideró en el año 2000 un sistema de legado, formado por muchas aplicaciones que eran difíciles y engorrosos de usar. ACS se construyó sobre muchas herramientas de software obsoletas de la década de 1970, incluido el lenguaje de programación natural, el ADABAS sistema de base de datos de gestión, y la terminal de pantalla verde de IBM 3270.

El software fue originalmente destinado a ser implementado a mediados de 2004, y fue originalmente destinado a ser poco más que una web de front-end a los datos existentes de ACS.

---

Análisis del caso:

Analizando el caso podemos decir que el proyecto fracasa rotundamente por distintas y variadas causas, como lo son “planificar con datos insuficientes” ya que la falta de un fuerte plan desde el principio llevó a las malas decisiones de arquitectura, “los requerimientos cambiantes” fue otro de los causantes del fracaso, “La falta de recursos” porque no se contaba con mano de obra especializada para desarrollar tal proyecto y por último “falta de involucramiento del usuario” ya que varias decisiones se llevaban a cabo por los desarrolladores sin consultar al product owner.

Por otro lado el caso se incluye dentro de 5 de las 6 estimaciones del CHAOS report 1. Requerimientos incompletos 13.1 %, Falta de involucramiento del usuario 12.4 % , Falta de recursos 10.6 %, Falta de apoyo de la Gerencia 8.7 %, Requerimientos cambiantes 8.1 %.

### **4.3. El proyecto Stretch de IBM**

International Business Machines Corporation (IBM), la compañía multinacional de tecnología y consultoría con sede en Nueva York, EE. UU. De modo indefectible alcanza los puestos más elevados de las listas de marcas mundiales.

A finales de la década de los cincuenta, más específicamente en 1956, un grupo de científicos informáticos de IBM se propuso diseñar y fabricar el ordenador más rápido y tecnológicamente avanzado del mundo: el superordenador IBM 7030 Stretch. Les tomó 5 años y muchas horas de trabajo, pero finalmente produjeron el IBM 7030 conocido cariñosamente como 'Stretch'.

El objetivo del proyecto era crear una computadora que superaría entre 100 y 200 veces la velocidad y el rendimiento de su más cercano competidor y, de ese modo, «alargaría» los límites del diseño de este tipo de máquinas. Un objetivo tan ambicioso e impresionante como éste hizo que su precio se fijara en 13,5 millones de dólares.

Stretch fue la supercomputadora más rápida del mundo y mantuvo este título hasta 1964 que entró en funcionamiento CDC 6600. ¿Capaz de manejar medio millón de instrucciones por segundo, seguramente pensaría que Stretch fue considerado un éxito por IBM? Bueno, te equivocarías.

---

El líder del proyecto, Stephen W. Dunwell, reconoció más tarde que el proyecto era tan complicado que «jamás antes habían tenido que funcionar simultáneamente tantas cosas en un único ordenador». Los ingenieros tuvieron que hacer frente a un montón de retos para diseñar y fabricar muchos de los elementos del innovador sistema. Entre ellos, un conmutador distribuidor de carga que permitiese utilizar transistores para accionar la memoria de núcleos de ferrita.

Las previsiones demasiado optimistas hicieron que se exceda de largo tanto el plazo de ejecución como los costes del proyecto. Asimismo, cuando a comienzos de los sesenta se probó la primera versión operativa del Stretch, tan sólo era 30-40 veces más rápido que su predecesor. Fue considerado un rotundo fracaso y el precio de los sistemas que ya habían sido pedidos se redujo a 7,78 millones de dólares, por debajo del precio de coste que era 13,5 millones de dólares. Solo se construyeron 9 computadoras Stretch y finalmente se retiraron del mercado.

Aun así, hubo un resquicio de esperanza. Las innovaciones en fabricación, presentación y arquitectura que Stretch aportó fueron la piedra angular de no pocos avances venideros para la compañía y la catapultó a la vanguardia de la industria. Si las expectativas no hubiesen sido tan elevadas, quizás el proyecto hubiera tenido éxito, pero el hecho es que a Stretch sólo le queda resignarse a figurar en los libros de Historia como un integrante más de listas de «fracasos de gestión de proyectos» como ésta.

#### Análisis del caso:

Estudiando el caso previamente descrito podemos apreciar que claramente el mismo se ve afectado por algunas de las causas comunes para proyectos fallidos las cuales son “Fallas al definir un problema” lo cual no se establecieron metas realistas, es decir que los objetivos del proyecto eran ambiciosos pero irreales. “Planificar con datos insuficientes” ya que se desconocía la información sobre el funcionamiento simultáneo de varios programas en un único ordenador. “Requerimientos incompletos o cambiantes” y “falta de recursos” ya que los ingenieros tuvieron que hacer frente a un montón de retos para diseñar y fabricar muchos de los elementos del innovador sistema. Podemos ubicar al caso dentro del 9,3% de los proyectos con expectativas poco realistas.

---

Si bien Stretch fue un fracaso, introdujo la canalización, la protección de la memoria, el intercalado de memoria y otras tecnologías que han dado forma al desarrollo de las computadoras como las conocemos hoy.

## **5. Ejemplos de proyectos exitosos**

### **5.1. Facturación de Energía Eléctrica**

Aplicación de bases de datos encargada de manejar el proceso de la facturación de usuario Regulados y No Regulados, proceso que anteriormente no era realizado en la empresa pues se subcontrataba un externo para que lo realizara lo cual era una desventaja para la empresa por que no contaba con los datos internamente para realizar una re facturación un duplicado de factura, recibir adecuadamente un reclamo o solicitud, puesto que se debía transmitir a otros y esperar un tiempo una respuesta.

Con el desarrollo del sistema de facturación la empresa logró mejorar su atención al cliente dándole información de forma rápida consistente y resolviendo la mayoría de los asuntos inmediatamente como un cambio de nombre, emisión de duplicados, correcciones de consumos entre otros.

Análisis del caso:

Estudiando el caso previamente descrito podemos apreciar que se cumplieron los puntos más importantes de un proyecto exitoso como el “monitoreo y feedback” ya que la implementación y desarrollo avanzó correctamente como se había planificado y además se realizó una retroalimentación adecuada. Los Directores del proyecto estuvieron siempre comprometidos con el eficiente desarrollo del proyecto.

Los proveedores elegidos fueron idóneos para cumplir con el objetivo del sistema respondiendo al punto de “tener un objetivo claro” y se logró correcta delimitación del proyecto.

Por ultimo se cumplio con el mas importante de todos que es la “comunicacion” ya que hubo un correcto entendimiento entre usuarios finales del sistema, directivos y desarrolladores para realizar una correcta captura de requerimientos. Los procesos de gestión fueron correctos en

---

cuanto a integración, alcance, tiempo, costo, calidad, recursos humanos, comunicaciones y riesgos logrando así un proyecto exitoso.

## **5.2. Spotify**

Otro caso de éxito es el de una empresa que ha sabido adaptarse perfectamente a las metodologías ágiles(muy utilizadas hoy en día) es Spotify, haciendo especial hincapié en la figura del Scrum Master. Muchas veces contratan un Agile Coach externo con una gran experiencia en el campo para liderar los proyectos. Vemos aquí la importancia de contar con roles especializados que conozcan las metodologías ágiles para llevar un proyecto de este tipo al éxito. Ya no solo el Scrum Master, sino también otros roles como el Product Owner, responsable de entender al cliente y al usuario para saber trasladar en tiempo y forma la información adecuada al equipo de desarrollo.

Spotify es consciente de la metodología de trabajo de su competencia (Google o Apple por ejemplo), por lo que decidieron acercarse al Scrum de forma muy sistemática. Compitiendo contra semejantes corporaciones, sabían que en cualquier momento podrían ser derrotados a menos que fuesen más rápidos, más baratos y mejores.

Por ejemplo, fijémonos en el nuevo iTunes Radio, ofrece exactamente lo mismo que Spotify. Es por eso que han tenido que mejorar sus equipos de trabajo para asegurarse que van más rápido. En Spotify los equipos se organizan por escuadrones (squads), pequeños equipos de Scrum con la habilidad de implementar el software desarrollado al final de cada sprint, sin romper ningún otro equipo. Una característica curiosa del funcionamiento de Spotify es que cada uno de estos pequeños grupos tiene una parte del producto que es totalmente suyo.

Aun así, Spotify necesita implementar, cambiar y actualizar su código constantemente sin romper nada más. Para ello es necesaria una buena coordinación central de la compañía.

Si quieres ser rápido también es necesario deshacerse de todas aquellas partes del proceso que entorpezcan el avance. En Spotify, por ejemplo, contaban con un equipo de operaciones que se encargaba de las implementaciones, pero el funcionamiento era demasiado lento. Por eso decidieron eliminar esta fase y hacer que los propios desarrolladores implementaron sus trabajos. En un negocio tan creciente como Spotify, surgen muchos impedimentos, Spotify

---

lidia con esos impedimentos monitoreando constantemente los obstáculos potenciales y eliminándolos antes de que estos se conviertan en problemas mayores.

Uno de los casos más documentados sobre las soluciones a sus problemas es el siguiente:

En una ocasión notaron que su equipo de operaciones estaba actuando muy lento para desplegar sus soluciones. Entonces decidieron eliminar ese impedimento transfiriendo la responsabilidad del despliegue de las soluciones al propio equipo de desarrollo. Los encargados de operaciones sólo apoyan con herramientas y procedimientos.

Esa decisión no suele tener sentido en otras organizaciones, sin embargo, ellos determinaron que debían ser pragmáticos, abrir la mente y ser disruptivos, así han logrado el éxito hasta hoy.

#### Análisis del caso:

El éxito de este proyecto de desarrollo se basa en algunos motivos que vale la pena resaltar como lo son “Monitoreo y feedback” y “Comunicación” debido a que en el equipo de desarrollo existía una buena fluidez de la información sobre agregados o cambios que eran informados adecuadamente a cada miembro, y a su constante y periódica revisión sobre el proyecto para evaluar si seguían por el camino correcto previamente planificado, cuestiones claves como estas son los que los ayudaron en el éxito de su proyecto. También cabe destacar el “objetivo claro” que tuvo desde el principio el equipo con lo que quería lograr con su producto teniendo muy bien definida su misión a largo plazo.

---

## CONCLUSIÓN

Las causas y motivos en los éxitos como fracasos de los proyectos de desarrollo de software nos marcan una clara idea sobre como emprender un proyecto, para no cometer los mismos errores que se comenten en miles de proyectos tenemos que tenerlos bien presentes y más si nuestra idea a futuro es la de emprender un proyecto de desarrollo de software exitoso, de los ejemplos observamos como realmente no cumplir con los ítems de fracaso y éxito te puede llevar a un resultado muy perjudicial económicamente como así también de reputación. Pudimos estudiar y comprender cómo la información dentro de los proyectos es fundamental para que todas las partes intervinientes estén debidamente informadas sobre las situaciones y cambios para evitar errores y un mal entendimiento de los requerimientos , para ello se debe contar con óptimos canales de información. El papel que cumple el recurso humano es prioritario para un proyecto exitoso , su capacitación como experiencia son piezas valiosas para cada una de los recursos así se forman equipos que trabajen en base a buenas prácticas dentro del proyecto de desarrollo.

## BIBLIOGRAFÍA

- [1.] Sommerville, Ian - INGENIERÍA DE SOFTWARE - Novena Edición (Editorial Addison-Wesley Año 2011). Capítulo 22, 23.
- [2.] Pressman, Roger - INGENIERÍA DE SOFTWARE, UN ENFOQUE PRÁCTICO. Séptima Edición - Editorial McGraw Hill – Año 2010. Capítulo 24
- [3.] <http://www.cmetrix.la/6-claves-para-un-proyecto-exitoso/>
- [4.] <https://blog.anfix.com/como-gestionar-proyecto-con-exito/>
- [5.] <https://www.evaluandoerp.com/diez-claves-para-el-exito-de-un-proyecto-informatico/>
- [6.] [https://www.coffeekids.org/wp-content/uploads/2013/10/Fase\\_Monitoreo-y-Evaluacion-de-proyectos\\_Victor-Garcia.pdf](https://www.coffeekids.org/wp-content/uploads/2013/10/Fase_Monitoreo-y-Evaluacion-de-proyectos_Victor-Garcia.pdf)
- [7.] [http://www.iso-21500.es/sites/default/files/ficheros\\_guia\\_iso21500/g\\_iso21500\\_rsk\\_c01\\_v01\\_el\\_caso\\_del\\_aeropuerto\\_de\\_denver.pdf](http://www.iso-21500.es/sites/default/files/ficheros_guia_iso21500/g_iso21500_rsk_c01_v01_el_caso_del_aeropuerto_de_denver.pdf)

- 
- [8.] <https://drich1145.wordpress.com/exitos-y-fracazos-de-un-proyecto-de-software/>
- [9.] <https://strategicppm.wordpress.com/2010/04/05/the-fbis-virtual-case-file-project-and-project-failure/>
- [10.] <http://ingenieriadesoftwareg4.blogspot.com/>
- [11.] <http://www.chilton-computing.org.uk/acl/literature/reports/p018.htm>
- [12.] <http://www.brouhaha.com/~eric/retrocomputing/ibm/stretch/>
- [13.] [https://www.ibm.com/ibm/history/exhibits/mainframe/mainframe\\_PP7030.html](https://www.ibm.com/ibm/history/exhibits/mainframe/mainframe_PP7030.html)
- [14.] <https://web.archive.org/web/20090715121354/http://cyclone.cs.clemson.edu/~mark/stretch.html>

### **Referencia de la plantilla utilizada en el informe técnico**

- [15.] [https://www.academia.edu/5091078/INFORME\\_T%C3%89CNICO\\_LA\\_ESTRUCTURA\\_DEL\\_INFORME\\_T%C3%89CNICO\\_EST%C3%81\\_FORMADA\\_POR\\_La\\_parte\\_inicial?auto=download](https://www.academia.edu/5091078/INFORME_T%C3%89CNICO_LA_ESTRUCTURA_DEL_INFORME_T%C3%89CNICO_EST%C3%81_FORMADA_POR_La_parte_inicial?auto=download)