

Caso

Proyecto Infinito

20 de septiembre de 2020

Eduardo Albarizaez – 1106918

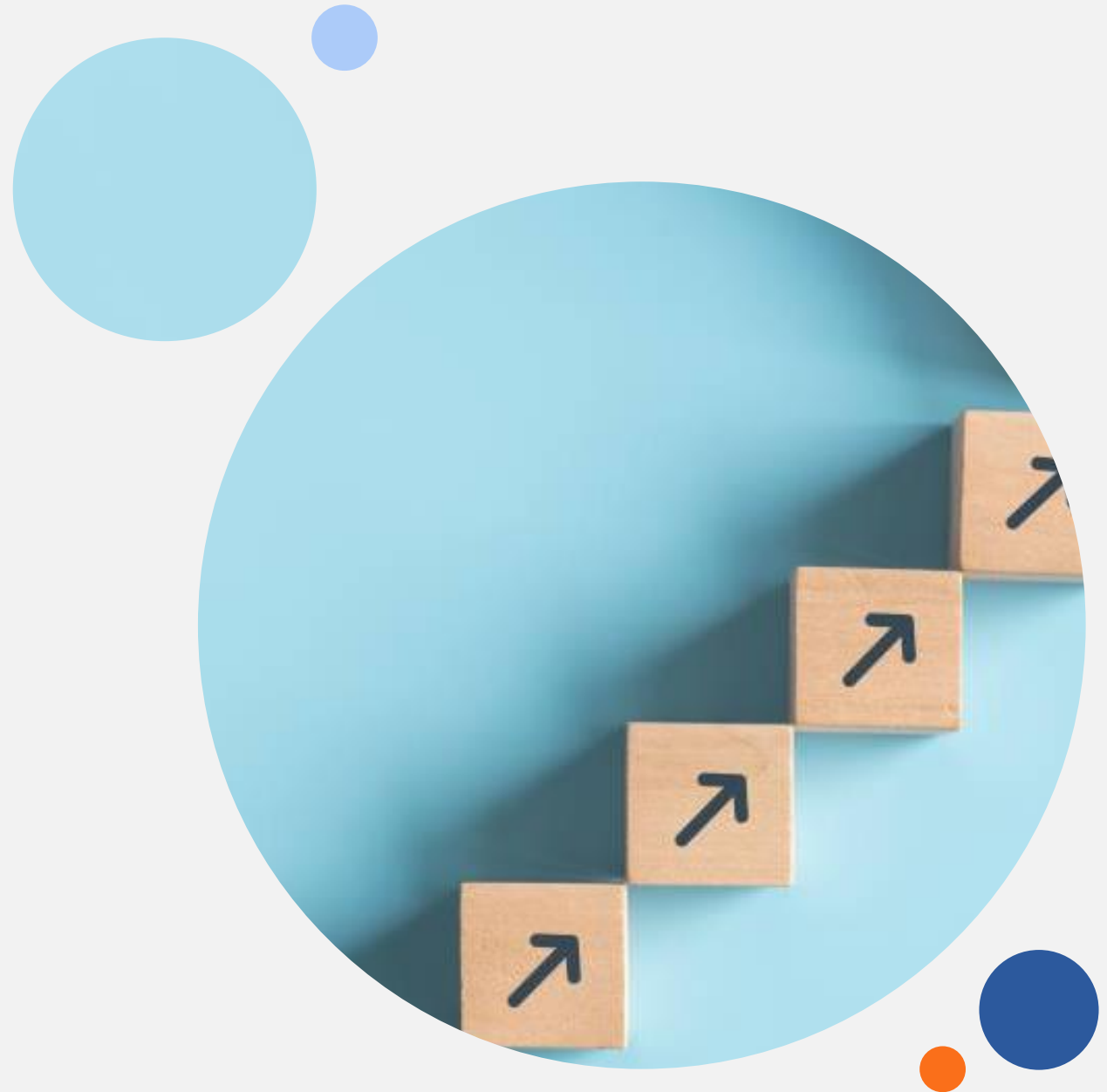
Marcos Calderón 1060918

Boris Morán – 1194518

Luis Ovalle - 1037214

Principios del software

- La razón del que exista todo
 - El crecimiento de la empresa esta generando problemas en la búsqueda de un sistema robusto y eficiente
- MSE
 - Pueden adaptar el software existente a nuevas tecnologías.
 - En caso de crear nuevo software se recomienda reutilizar componentes.



Principios del software

- **Mantener la visión**
 - Los cambios efectuados en la manera de realizar pruebas en los componentes.
 - Cambio drástico de líder.
 - Contratación de nuevos programadores.
- **Otros consumirán lo que produce**
 - Las pruebas fueron realizadas por usuarios no enterados del proyecto, en su totalidad.
 - Pedían cambios al proyecto inicial.





Principios del software

- **Ábrase al futuro**
 - El proyecto surgió por crecimiento significativo.
 - No se planificó que el nuevo software fuese versátil a nuevos cambios.
- **Planee por anticipado la reutilización**
 - Sí se planificó la reutilización de componentes.
 - En el cambio de roles se perdió el conocimiento de estos.
 - Creaban nuevos módulos sin funcionar.



Mitos del software

Proyecto Infinito



Mitos de la administración

- El nuevo líder agregó más programadores. Forzando, a los que estaban desarrollando, a introducir a los nuevos al código en desarrollo. Atrasando el proyecto.
- “Contrato a una empresa externa para el desarrollo, puedo descansar y que ellos lo elaboren”. El administrador y asistente de sistemas no estaban capacitados para subcontratar un proyecto de esta magnitud.





Mitos del cliente

- Los requerimientos del software cambian continuamente. Los nuevos usuarios, continuamente, estaban solicitando cambios no previstos.

Mitos del profesional

- Una vez escribamos el programa y funciona, ya terminó el trabajo. Esto ya que a los programadores solo se les exigía la entrega de código sin tomar en cuenta el tiempo y planificación.



Fases estructurales del proceso de software

Actividades estructurales



Comunicación

- Esta etapa la cumplieron ya que se llevo a cabo un análisis de requerimientos de varios meses.
- En un cronograma de 1 año se llevaron “varios meses” en esta fase.
- Esta fase pierde valor al momento de los cambios inesperados, solicitados por los nuevos usuarios.

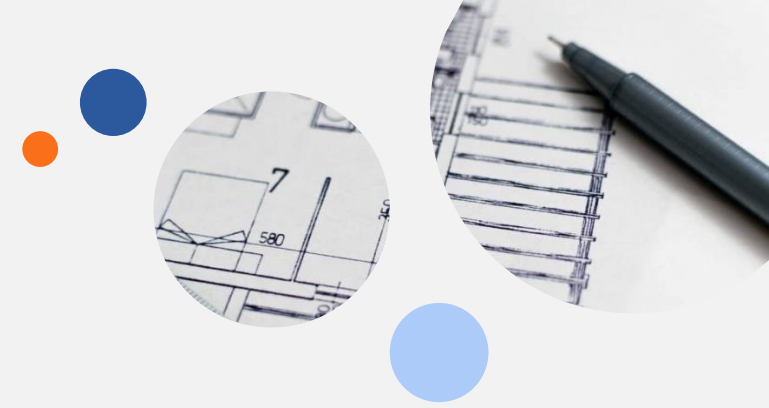
Planeación

- En la presentación del caso no hace mayor énfasis en esta fase. Pero se observa que luego del análisis de requerimientos siguieron con la generación de prototipos.
- Sin esta etapa se pudo observar que el proyecto no tenía mapa alguno para guiarse.
- Esta etapa, con un cronograma, ahorraría tiempo en otras etapas.

Modelado

- No se tuvo una visión de panorama general para encajar cada etapa del proyecto.
- No realizaron un entendimiento del proyecto por medio de algún modelo, como lo podría ser un diagrama de flujo.

Actividades estructurales

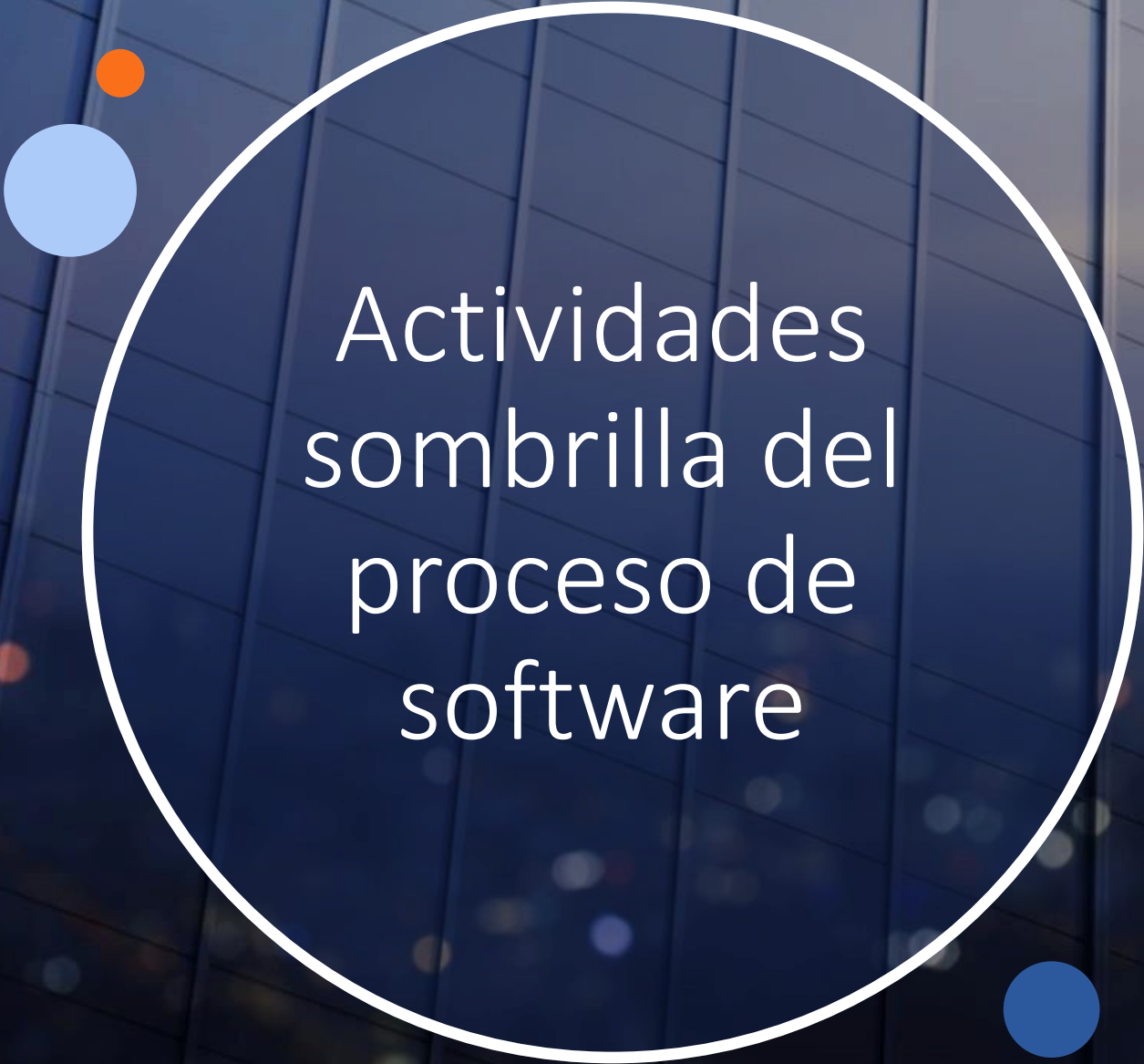



Construcción

- Al momento de reutilizar componentes, sin tener algún mapa que los guiara, esto hizo que no se generara software totalmente flexible a cambios.
- La falta de realización de pruebas genera caos y poca confiabilidad respecto al proyecto.

Despliegue

- Todas las pruebas finales, terminaban en fallos. Esto atrasaba el proyecto y se perdía credibilidad.
- El despido del departamento de QA causó que el software no fuera resiliente

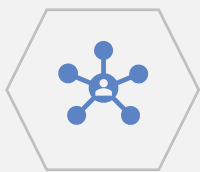


Actividades sombra del proceso de software



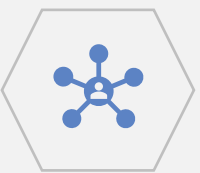
Seguimiento y control del proyecto de software

El líder del proyecto no garantizó que se cumplieran los requisitos y seguimiento de planificación a cabalidad.



Administración del riesgo

No se consideraron variables que ocasionaran atraso. Como lo fue la enfermedad del ingeniero de software y el proceso de adaptación de los desarrolladores junior.



Aseguramiento de la calidad del software

Rompieron con el 7mo principio ya que no pensaron en que consecuencias iba a tener el despedir al departamento de QA.



Administración de la reutilización

Se planificó desde un inicio y se inicio su implementación, pero no se tomó en cuenta si era correcta la reutilización, ni que se iban a unir nuevos desarrolladores.



Revisiones técnicas

Con el fin de agilizar la entrega, esta actividad nunca se llevo a cabo en la entrega de módulos fallidos.



Medición

Se debía medir el avance y calidad a lo largo del proyecto. Sin embargo, solo se desarrollaron módulos sin algún tipo de supervisión.



Administración de la configuración del software

Con una buena planificación se hubiera afrontado el diferente versionamiento que existía entre Dev, QA y producción, para que se trabajara en sintonía.



Preparación y producción del producto de trabajo

La presencia de un arquitecto para poder preparar modelos y poder eliminar ciertas inconsistencias en diferentes ambientes.