

## Hito 1 Ingeniería de software


- **Integrantes Actualizados:**

Diego Morales, 202073511-5

Nicolás Olivos, 202073507-7

Matías Sandoval, 202104684-4


- **Autoevaluación del estado del proyecto:**

 **Opportunity**

Value Established


- ✗ Opportunity value quantified
- ✓ Solution impact understood
- ✓ System value understood
- ✓ Success criteria clear
- ✗ Outcomes clear and quantified

3 / 6

 **IVAR JACOBSON**  
INTERNATIONAL  
Generated by IJ Practice Workbench™

1.1.2


Nos encontramos en el nivel "Solution needed", ya que, durante el desarrollo del software nunca se cuantificó el costo oportunidad que conllevaría el esfuerzo y los resultados del software. Sin embargo, para los nuevos requerimientos esperamos darle mayor relevancia a lo que vamos a desarrollar para el producto final.

 **Stakeholders**

In Agreement

- ✓ Minimal expectations agreed
- ✗ Rep's happy with their involvement
- ✗ Rep's input valued
- ✗ Team's input valued
- ✓ Priorities clear & perspectives balanced

4 / 6

 **IVAR JACOBSON**  
INTERNATIONAL  
Generated by IJ Practice Workbench™

1.1.2

Nos encontramos en el nivel "Involved", porque, lo largo del desarrollo del software, el cliente se vio con mucho interés en aquel desarrollo, mediante sus requerimientos vistos en clases. Pero actualmente no sabemos si el cliente ha probado el software para ver si satisface todas sus necesidades con la versión actual del proyecto. Como equipo aun no hemos probado todos los casos bordes que puedan existir mediante los datos a evaluar.



## Requirements

### Acceptable

- ✓ Acceptable solution described
- ✗ Change under control
- ✓ Value to be realized clear
- ✓ Clear how opportunity addressed
- ✓ Testable

4 / 6



IVAR JACOBSON  
INTERNATIONAL  
Generated by I4 Practice Workbench™

1.1.2

Nos encontramos en el nivel "Coherent", debido a que, los requerimientos principales dados por el cliente fueron reescritos a historias de usuarios para representar de mejor manera como funcionara el sistema, para posteriormente ser programados, sin embargo, actualmente una de las funcionalidades dentro del software que realiza cambios, necesita ser actualizada/reparada, ya que, afectaría en el futuro la eficiencia del software.



## Software System

### Ready

- ✗ User documentation available
- ✓ System accepted as fit-for-purpose
- ✓ Stakeholders want the system
- ✗ Operational support in place

4 / 6



IVAR JACOBSON  
INTERNATIONAL  
Generated by I4 Practice Workbench™

1.1.2

Nos encontramos en el nivel "Usable", ya que, durante el desarrollo del software, documentamos los avances del sistema mensualmente en un repositorio para los integrantes del equipo de desarrolladores, sin embargo, no se realizó una documentación para los usuarios. Además, carecemos de un equipo operativo dedicado para brindar soporte en caso de cualquier eventualidad o problema relacionado con el sistema.



## Work

### Under Control

- ✓ Tasks being completed
- ✓ Unplanned work under control
- ✗ Risks under control
- ✓ Estimates revised to reflect performance
- ✓ Progress measured
- ✓ Re-work under control
- ✓ Commitments consistently met

4 / 6



IVAR JACOBSON  
INTERNATIONAL  
Generated by IJ Practice Workbench™

1.1.2

Nos encontramos en el nivel "Started", pues, a lo largo del desarrollo del software, como equipo consideramos los riesgos principales para el cliente. Sin embargo, no todos los riesgos están bajo control dentro del software, ya que, una de las funcionalidades que realiza cambios, no es del todo eficiente. Un riesgo que considerar para el futuro del software, debido a que no sería tan fácil de actualizar



## Way of Working

### In Place

- ✗ Used by whole team
- ✗ Accessible to whole team
- ✗ Inspected and adapted by whole team


4 / 6



IVAR JACOBSON  
INTERNATIONAL  
Generated by IJ Practice Workbench™

1.1.2

Nos encontramos en el nivel "In use", ya que el software aún no ha sido implementado por todo el equipo. En consecuencia, el equipo sigue en proceso de adaptación y estableciendo una comunicación continua para garantizar una transición efectiva hacia la plena utilización del software.


**Team**

Collaborating


✖ Works as one unit

✔ Communication open and honest

✔ Focused on mission

✖ Members know each other

3 / 5


**IVAR JACOBSON**  
INTERNATIONAL  
Generated by UI Practice Workbench™

1.1.2

Nos encontramos en el nivel "Formed", debido a que, en retrospectiva al software desarrollado, aunque mantuvimos una comunicación sólida, enfrentamos desafíos para trabajar como una sola unidad. Surgieron muchos problemas debido a nuestra falta de conocimiento. Sin embargo, esta experiencia nos enseñó la importancia de fortalecer nuestros lazos como equipo para futuros proyectos, lo que seguramente con el nuevo equipo nos llevará a una colaboración más efectiva y resultados exitosos.

- **Alphas Prioritarios:**

1- **Requerimientos:** Si bien el software es funcional, aún existe problema a la hora de editar una hora dentro del calendario, que al estar implementado como una concatenación implícita de eliminar los datos en la hora seleccionada y agregar los nuevos datos que se rellenan en el formulario emergente en vez de una edición de los datos ya existentes, hace que disminuya la eficiencia del software, por lo que se requiere un cambio en la implementación. Además, existen problemas menores al respecto de la interfaz. Para el primer problema, simplemente hay que mejorar el uso de las operaciones CRUD, así optimizando el rendimiento, mientras que para el otro hay que enfocarse en lo que se refiere al diseño del software de manera que sea lo más cómodo posible para el cliente.

2- **Equipo:** Dado a la incidencia de un nuevo integrante dentro del grupo, es importante priorizar la integración de este dentro del equipo y del proyecto, pues es fundamental que todos los miembros del equipo puedan colaborar y brindar soporte para poder concretar de manera eficiente los objetivos propuestos. Para poder pasar al siguiente estado, hay que contextualizar lo antes posible al nuevo integrante al grupo, al código, a los objetivos y al funcionamiento del software.

- **Riesgos identificados:**

- 1- Muchas horas medicas en un mismo día: Cuando se agendan muchas horas medicas para el mismo día, el calendario que vemos en el software puede verse afectado de tal manera que visualmente no es lo adecuado, no es atractivo para el cliente.
  - Impacto: Alto
  - Probabilidad de ocurrencia: Medio
  - Medidas de mitigación: (Riesgo activo y no mitigado) Una solución propuesta para abordar este problema es implementar la agrupación de citas programadas para el mismo día. Se establecería una condición que, si hay más de tres citas programadas, estas se agruparían visualmente para evitar que afecten la presentación del calendario.
- 2- Horas medicas con la misma hora y tipo de examen: Si se agendan ciertas horas medicas al mismo tiempo en el mismo día, hora y tipo de examen, el software no está preparado para tener una especie de seguridad o bloqueo para resolver esos casos.
  - Impacto: Alto
  - Probabilidad de ocurrencia: Bajo
  - Medidas de mitigación: (Riesgo activo y no mitigado) Una de las ideas a implementar es hacer un bloqueo a nivel de usuario, al momento de agendar una hora médica, en el transcurso de rellenar el formulario estará bloqueado el evento de agendar una hora médica para los demás usuarios hasta que se le de a guardar, esto desbloqueara la función y la podrán usar los demás usuarios, así evitando un guardado simultaneo de horas médicas.
- 3- Editar una hora medica: El botón de editar una hora medica realmente funciona como el botón de eliminar una hora medica y a la vez crear una nueva hora médica, por lo tanto, está realizando una doble función en vez de ser solo una, afectando a la eficiencia del software.
  - Impacto: Medio
  - Probabilidad de ocurrencia: Bajo
  - Medidas de mitigación: (Riesgo activo y no mitigado) Implementar correctamente el método CRUD para el sistema, para que funcione correctamente el botón editar.