

Tarea N°2 - ELO 329 :
Descuentos, Recomendaciones y Aspectos
Destacados

Jesús Márquez - Nicolas Olivos

Índice

I	Recomendaciones Generales	2
	I-A Respecto al Código de la Tarea	2
	I-B Respecto a la Entrega y Documentación de la Tarea	2
II	Grupo 1 : 201330011-1 - 201330045-6 201303018-1	2
	II-A Descuentos	2
	II-B Recomendaciones	2
	II-C Aspectos Destacados	2
III	Grupo 2 : 201430052-2 - 201430035-2 - 201430002-6	3
	III-A Descuentos	3
	III-B Recomendaciones	3
	III-C Aspectos Destacados	3
IV	Grupo 3 :201430028-K - 201330006-5 - 201330057-K	3
	IV-A Descuentos	3
	IV-B Recomendaciones	3
	IV-C Aspectos Destacados	3
V	Grupo 4 :201530017-8 - 201504012-5 - 201354052-K	4
	V-A Descuentos	4
	V-B Recomendaciones	4
VI	Grupo 5 :201203030-7 - 201321071-6	4
	VI-A Descuentos	4
	VI-B Recomendaciones	4
	VI-C Aspectos Destacados	4
VII	Grupo 6 :201321036-8 - 201203028-5	4
	VII-A Descuentos	4
	VII-B Recomendaciones	4
VIII	Grupo 7 :201204008-6 - 201221011-9 - 201221003-8	5
	VIII-A Descuentos	5
	VIII-B Recomendaciones	5
	VIII-C Aspectos Destacados	5
IX	Grupo 8 :201121009-3 - 201321003-1 - 201223029-2	5
	IX-A Descuentos	5
	IX-B Recomendaciones	5
	IX-C Aspectos Destacados	5
X	Grupo 9 : 201321032-5 - 2903012-K - 201121023-9	6
	X-A Descuentos	6
	X-B Recomendaciones	6
	X-C Aspectos Destacados	6
XI	Grupo 10 :2921049-7 - 201021012-K	6
	XI-A Aspectos Destacados	6

XII	Grupo 11 : 201121030-1 - 201204753-6	7
	XII-A Descuentos	7
	XII-B Recomendaciones	7
	XII-C Aspectos Destacados	7
XIII	Grupo 12 :201321023-6 - 201321006-6 - 201321025-2	7
	XIII-A Descuentos	7
	XIII-B Recomendaciones	7
XIV	Grupo 13 :201530004-6 - 201403016-9 - 201530009-7	8
	XIV-A Descuentos	8
	XIV-B Recomendaciones	8
	XIV-C Aspectos Destacados	8
XV	Grupo 14 :201221075-5 - 201221031-3	8
	XV-A Descuentos	8
	XV-B Recomendaciones	8
	XV-C Aspectos Destacados	8
XVI	Grupo 15 : 201430003-4 - 201404583-2 - 201430001-8	9
	XVI-A Descuentos	9
	XVI-B Recomendaciones	9
	XVI-C Aspectos Destacados	9
XVII	Grupo 16 :201530008-9 - 201530001-1 - 201530002-k	9
	XVII-A Descuentos	9
	XVII-B Recomendaciones	9
	XVII-C Aspectos Destacados	9
XVIII	Grupo 17 : 201551010-5 - 201530024-0 - 201630015-5	10
	XVIII-A Descuentos	10
	XVIII-B Recomendaciones	10
XIX	Grupo 18 :201130007-6 - 201130004-1	10
	XIX-A Descuentos	10
	XIX-B Recomendaciones	10
XX	Grupo 19 :201103003-6	11
	XX-A Descuentos	11
	XX-B Recomendaciones	11
XXI	Grupo 20 :201221024-0 201121047-6 - 201104136-4	11
	XXI-A Descuentos	11
	XXI-B Recomendaciones	11
XXII	Grupo 21 : 201790126-8 - 201321060-0	12
	XXII-A Descuentos	12
	XXII-B Recomendaciones	12
	XXII-C Aspectos Destacados	12

I. RECOMENDACIONES GENERALES

I-A. *Respecto al Código de la Tarea*

- Todo supuesto que hagan a la hora de programar su solución, se deben comentar en la documentación de la tarea para tener conocimiento de aquello, también es valido preguntarlo por correo, para que haya constancia de lo que están pensando a la hora de programar.
- Evitar el "HardCodeo".
- eliminar los comentarios que se dejan como prueba (Ej: print, //to be coded, etc) ya que desordena el código y la lectura no es fácil.

I-B. *Respecto a la Entrega y Documentación de la Tarea*

- Cumplir con el formato solicitado en las normas de entrega de tareas.
- No agregar archivos .class, .csv git ignore ni similares en las entregas.
- Generar un makefile por etapa, no es necesario generar un makefile que compile todas las etapas a la vez.
- Mantener la formalidad en comentarios dentro del código y en los commits realizados en el repositorio GIT.
- Subir paulatinamente el avance del código en GIT, no subir etapas completas. La idea es tener un respaldo de los cambios realizados en caso de que alguna modificación perjudique el funcionamiento del programa y no se logre visualizar la fuente. Esto además permite el uso correcto de la plataforma que es el desarrollo constante de código para poder ayudar a los demás colaboradores.
- A la hora de llegar a una solución propuesta que probablemente sera su solución final, es imperioso que lo prueben en el servidor de aragorn puesto que ahí es donde revisamos las tareas.

II. GRUPO 1 : 201330011-1 - 201330045-6 201303018-1

II-A. *Descuentos*

- No se hace uso adecuado de JavaDoc al no utilizar rótulos para parámetros y variables a retornar (**-2 Puntos a Uso JavaDoc**).
- En etapa 3, botones de la caja ascensor no reflejan si han sido presionados. Además no se coloca un indicador de piso en la botonera de la cabina (**-4 Puntos a Etapa 3**).
- En etapa 3, la implementación del JSplitPane no permite aprovechar las ventajas que entrega este tipo de contenedor, además de no respetar la distribución mostrada en la imagen de ejemplo del enunciado de la tarea (**-3 Puntos a Etapa 3**).
- Menús de Velocidad y Tiempo de Espera no se encuentran operativos (**-12 Puntos a Etapa 4**).

II-B. *Recomendaciones*

- No agregar commits con las mismas descripciones, tratar de ser más específicos cuando se realizan arreglos y tratar de utilizar un solo idioma para los commits del proyecto.
- Indicar como se solucionaron las dificultades encontradas en la tarea.
- Eliminar comentarios tipo *to be coded* al finalizar el código.
- Al realizar el pintado de los sensores, el cambio de color se hace dentro de la vista de caja ascensor. Se recomienda que esto se haga dentro de la vista del sensor en sí y que la caja del ascensor solo se encargue de llamar al dibujado del elemento. Lo mismo con la cabina, el llenado debiese realizarse dentro de la vista de cabina, no en la caja del ascensor.
- Tener cuidado con atributos no inicializados, como ocurre en clase Botón de la etapa 3, éstos pueden generar errores de referencias y punteros a nulo.
- Se recomienda generar en carpeta aparte la documentación de javadoc.

II-C. *Aspectos Destacados*

- El uso del formato *markdown* para la redacción del archivo *Readme* y el formato de presentación de éste permiten una explicación sencilla pero precisa del funcionamiento de los programas de las tareas.

III. GRUPO 2 : 201430052-2 - 201430035-2 - 201430002-6

III-A. Descuentos

- 201430052-2 y 201430035-2 tienen muchos comentarios con información que no aporta al proyecto y/o es de un lenguaje no apto para referirse a commits. Se incluyen archivos no relacionados con la tarea que se consideran archivos basura. **(-2 Puntos a Uso de Repositorio GIT)**.
- en la etapa 3 se encuentra un error de ejecución que si estoy en un piso X y apreté el botón del piso para bajar o subir el ascensor baje al piso anterior y luego llega al piso solicitado siendo de que siempre estuvo ahí **(-5 Puntos a Etapa 3)**.
- error supuestamente corregido según los commits no fue corregido el ascensor se va al inframundo **(-5 Puntos a Etapa 3)**.
- no se hace uso de JSplitPanel etapa 3 **(-5 Puntos a Etapa 3)**.
- etapa 4 mismo error anterior **(-15 Puntos a Etapa 4)**.

III-B. Recomendaciones

- en la etapa 3 la vista no queda bien visualizada a la hora de ejecutar, para esto se tuvo que agrandar la ventana, les recomiendo antes de entregar todo ejecutar el código en argon puesto que los ayudantes revisamos desde ahí.

III-C. Aspectos Destacados

- Cabe destacar el orden de presentación de las cosas, además el código bien indentado.

IV. GRUPO 3 :201430028-K - 201330006-5 - 201330057-K

IV-A. Descuentos

- No se incorporan diagramas de clases a la documentación **(-4 Puntos a Diagramas de Clases)**.
- 201430028-K genera muchos tags que aluden al final de la tarea, además de realizar el tag *Final* en medio del proyecto. 201330057-K solo hace 1 commit y no se considera como un avance significativo para el proyecto. 201330006-5 realiza demasiados commits con la misma descripción, no entregando una idea clara de lo aportado al proyecto **(-2 Puntos a Uso de Repositorio GIT)**.
- No se hace uso adecuado de JavaDoc al no generarse correctamente en servidor de argon por presencia de tilde y otros caracteres no permitidos. **(-6 Puntos a Uso JavaDoc)**.
- En etapa 3, la implementación del JSplitPane no permite aprovechar las ventajas que entrega este tipo de contenedor **(-3 Puntos a Etapa 3)**.

IV-B. Recomendaciones

- Explicar en el archivo *readme* como ocupar todos los rótulos presentes del archivo *Makefile*
- Tener más precaución con el uso de los tags, no generar el tag *final* si aún no se termina de editar la tarea. En futuras entregas se considerarán los archivos de dicho commits y no los últimos subidos.
- A pesar de generar una documentación íntegra, no se recomienda documentar todas las clases en el formato JavaDoc para estas tareas, basta con comentar las clases solicitadas, para que la tarea no demande más tiempo del que es necesario.
- La impresión realizada de la posición de la caja ascensor en la etapa 2 es innecesaria y solo satura de información el bash.
- Al realizar el pintado de los sensores, el cambio de color se hace dentro de la vista de caja ascensor. Se recomienda que esto se haga dentro de la vista del sensor en sí y que la caja del ascensor solo se encargue de llamar al dibujado del elemento. Lo mismo con la cabina, el llenado debiese realizarse dentro de la vista de cabina, no en la caja del ascensor.
- La interfaz utilizada para modificar los parámetros *speed* y *pause time* es poco intuitiva, no deja claro si se realiza la modificación. Tratar de dejar algún mensaje o señal para indicar cambios.
- Se recomienda generar en carpeta aparte la documentación de javadoc.
- Se recomienda probar todos los casos de uso de la tarea, para evitar errores como la no generación de documentación debido a tildes.
- Eliminar comentarios tipo *to be coded* al finalizar el código.

IV-C. Aspectos Destacados

- El formato de la documentación es agradable a la vista y la forma en que se presentan las dificultades y soluciones es sencilla pero precisa.
- En la etapa 1, la calidad de los comentarios del código es asombrosa.

V. GRUPO 4 :201530017-8 - 201504012-5 - 201354052-K

V-A. Descuentos

- 201530017-8 tienen muchos comentarios con información que no aporta al proyecto. 201354052-K tienen comentarios con información que no aporta al proyecto. 201504012-5 no existen commits con esta cuenta en git elo (**-2 Puntos a Uso de Repositorio GIT**).
- no se encuentra la clase interna BotoneraCabinaView documentada o no se genera bien (**-3 Puntos a Uso JavaDoc**).
- si se ponen parametros de referencia en la etapa 4 por ejemplo 0-15 deberian validar de que no se ingresen valores que sobrepasen esos valores, lo cual hace que el programa no funcione correctamente (**-5 Puntos a Etapa 4**).

V-B. Recomendaciones

- no utilizar caracteres especiales para la documentacion puesto que no queda bien expresada algunas palabras.
- a la hora de documentar y/o poner comentarios recordar dejar los comentarios que aporten informacion al codigo y no comentar las funciones que no sirven en ese momento sino que simplemente eliminarlas.

VI. GRUPO 5 :201203030-7 - 201321071-6

VI-A. Descuentos

- No se incorporan diagramas de clases de las etapas 1, 2 y 3 a la documentación (**-3 Puntos a Diagramas de Clases**).
- No se entrega pdf de documentación (**-4 Puntos a Entrega de Documentación General**).

VI-B. Recomendaciones

- Tratar de añadir la información justa y necesaria al readme, cosas como las dificultades y soluciones indicarlas en archivo pdf de documentación.
- En la primera etapa, la visualización de la clase botonera piso intermedio no es agradable a la vista, quizás estando el indicador de piso por sobre los botones se hubiera obtenido un mejor resultado.
- Al realizar el pintado de los sensores, el cambio de color se hace dentro de la vista de caja ascensor. Se recomienda que esto se haga dentro de la vista del sensor en sí y que la caja del ascensor solo se encargue de llamar al dibujado del elemento. Lo mismo con la cabina, el llenado debiese realizarse dentro de la vista de cabina, no en la caja del ascensor.

VI-C. Aspectos Destacados

- El formato del archivo readme es bastante agradable a la vista.
- El uso del repositorio Git cumple con lo esperado, se hace una cantidad apropiada de commits y los mensajes asociados a éstos entregan información relevante para el proyecto.

VII. GRUPO 6 :201321036-8 - 201203028-5

VII-A. Descuentos

- No existe redaccion de la documentacion lo cual enreda mas de lo que quiere explicar la documentacion (**-2 Punto a Entrega de Documentación General**).
- no se encuentra la clase interna BotoneraCabinaView documentada o no se genera bien (**-3 Puntos a Uso JavaDoc**).
- algunos metodos no estan especificados y/o bien documentados (**-1 Puntos a Uso JavaDoc**).
- no hubo uso constante de git solo en la semana final antes de la entrega. Se incluyen archivos no relacionados con la tarea que se consideran archivos basura (**-3 Puntos a Uso de Repositorio GIT**).
- se encuentra un error de ejecucion a la hora de apretar un boton de la cabina ya estando en ese piso queda en un bucle infinito lo cual pega el programa hasta que se activa algun boton de la botonoera de los pisos (**-5 Puntos a Etapa 3**).
- mismo error anterior etapa 4 (**-5 Puntos a Etapa 4**).

VII-B. Recomendaciones

- a la hora de programar probar los codigos que uds ya encuentran como solucion en aragorn donde los ayudantes probamos los codigos.

VIII. GRUPO 7 :201204008-6 - 201221011-9 - 201221003-8

VIII-A. Descuentos

- No se incorporan diagramas de clases de las etapas 2 y 3 a la documentación (-2 Puntos a Diagramas de Clases).
- En etapa 3, no se hace implementación del JSplitPane (-5 Puntos a Etapa 3).

VIII-B. Recomendaciones

- Tratar de cuidar la ortografía en la redacción del archivo de documentación y del readme.
- No añadir archivos extra a la entrega de la tarea.
- Eliminar comentarios tipo *to be coded* al finalizar el código.

VIII-C. Aspectos Destacados

- Se hace muy buen uso del repositorio GIT, dejando claro los cambios realizados y cada integrante aportando adecuadamente para el avance del proyecto.
- Tanto el readme como el archivo de documentación poseen un formato muy agradable a la vista.
- La forma en que se aborda la etapa extra es simple, pero poderosa.
- Es una muy buena idea separa el código de las imagenes a utilizar.

IX. GRUPO 8 :201121009-3 - 201321003-1 - 201223029-2

IX-A. Descuentos

- se incluyen archivos no relacionados con la tarea que se consideran archivos basura (-1 Puntos a Uso de Repositorio GIT)
- el ascensor en un momento dado se dispara hacia abajo y nunca vuelve suspendiendo la funcionalidad completa del programa (-15 Puntos a Etapa 4).
- algunas funciones no fueron para nada documentadas, lo que pierde el sentido el ejecutar java doc sobre estas (-2 Puntos a Uso de JavaDoc).
- la tarea pide específicamente que la documentación quede guardada en una carpeta "Documentacion" lo cual no se cumplió (-1 Puntos a Uso de JavaDoc).

IX-B. Recomendaciones

- cuando generen una solución que crean que es la correcta verifiquen si les funciona de manera completa en aragorn.

IX-C. Aspectos Destacados

- excelente informe, bien detallado y especificado, un excelente manejo de makefile.

X. GRUPO 9 : 201321032-5 - 2903012-K - 201121023-9

X-A. Descuentos

- No se incorporan diagramas de clases a la documentación (**-4 Puntos a Diagramas de Clases**).
- 201321032-5 no genera ningun commit, por lo que no se le entregan puntos por falta de uso del repositorio. 2903012-k solo hace 1 commit y no se considera como un avance significativo para el proyecto. 201121023-9 realiza muchos commits con una misma descripción, no entregando una idea clara del aporte concreto al proyecto. (**-2 Punto a Uso de Repositorio GIT**).
- No se hace uso completo de JavaDoc al no comentar todos los métodos ni todos los atributos de las clases solicitadas. (**-1 Punto a Uso JavaDoc**).
- En etapa 3, no se hace implementación del JSplitPane (**-5 Puntos a Etapa 3**).

X-B. Recomendaciones

- No generar atributos que no serán utilizados.
- En la etapa 3, la visualización de la botonera cabina no es agradable a la vista, lo ideal es tratar de imitar la visualización presentada en el enunciado.
- No realizar hardcoding para actualizar el estado de las clases.
- Eliminar comentarios tipo *to be coded* al finalizar el código.
- A pesar de generar una documentación integra, no se recomienda documentar todas las clases en el formato JavaDoc para estas tareas, basta con comentar las clases solicitadas, para que la tarea no demande más tiempo del que es necesario.
- Evitar if-else muy extensos para asignar imagenes, se pueden dar nombres genéricos a la imágenes, de forma que permitan ser utilizados en una iteración, por ejemplo *"1.png"*.
- La clase motor no debiese tener atributos relacionados con la interfaz gráfica, porque no se estaría cumpliendo la arquitectura Modelo Vista Controlador.
- El uso de la excepción por división por cero es una solución poco robusta, se recomienda crear clases que respondan a una excepción más específica.
- Al solicitar datos, especificar en que unidades se deben entregar.

X-C. Aspectos Destacados

- Archivo Readme entrega la información justa y necesaria.
- La documentación generada por JavaDoc es muy completa e integra.

XI. GRUPO 10 :2921049-7 - 201021012-K

XI-A. Aspectos Destacados

- excelente informe todo bien explicado.
- excelente solución de la tarea.

XII. GRUPO 11 : 201121030-1 - 201204753-6

XII-A. Descuentos

- En etapa 1, no se colocan un indicadores de piso en las botoneras mostradas (-2 Puntos a Etapa 1).
- En etapa 3, no se coloca un indicador de piso en la botonera de la cabina (-2 Puntos a Etapa 3).
- La lógica implementada para controlar la luz de la cabina no es robusta, lo que conlleva a que el ascensor realice viajes con la luz apagada. (3 Puntos a Etapa Extra)

XII-B. Recomendaciones

- Diseñar una portada al archivo de documentación.
- Indicar claramente las unidades de las variables a solicitar.
- Al realizar el pintado de los sensores, el cambio de color se hace dentro de la vista de caja ascensor. Se recomienda que esto se haga dentro de la vista del sensor en sí y que la caja del ascensor solo se encargue de llamar al dibujado del elemento. Lo mismo con la cabina, el llenado debiese realizarse dentro de la vista de cabina, no en la caja del ascensor.
- La clase sensor no debiese cambiar los colores de su visualización, porque no se estaría cumpliendo la arquitectura Modelo Vista Controlador. La clase Sensor.View debe ser la encargada de dicha labor.
- En la etapa 4, en vez de almacenar el motor como atributo para la clase *MyElevatorFrame* para modificar sus parámetros, asociar el listener del JMenú al motor directamente.

XII-C. Aspectos Destacados

- El formato de la documentación es bastante pulcro y ordenado, facilitando su lectura y entendimiento.
- Es una muy buena idea separa el código de las imagenes a utilizar.

XIII. GRUPO 12 :201321023-6 - 201321006-6 - 201321025-2

XIII-A. Descuentos

- despues de haber dado las condiciones de la tarea en las observaciones anteriores el formato del titulo del proyecto no calza con lo solicitado (1 punto Descuentos Varios).
- de parte del grupo no hubo un constante de la plataforma de git, solo hubieron unos comentarios que no reflejan el sentido del uso del repositorio (-3 puntos Uso de Git).
- falta la documentacion de la clase interna botoneraCabinaView (-3 puntos Uso de JavaDoc).
- en la etapa 3 no se considera apagar el boton de la cabina si apreto un boton del piso en cual estoy (-2 puntos Etapa 3).
- a la hora de cambiar la velocidad el ascensor no se detiene en ningun piso y deja de funcionar correctamente en la etapa 4 (-10 puntos Etapa 4).

XIII-B. Recomendaciones

- cuando lean el problema, de lo que se les pide lean bien o que realmente se les pide y no largarse a programar de inmediato.

XIV. GRUPO 13 :201530004-6 - 201403016-9 - 201530009-7

XIV-A. Descuentos

- No se incorporan diagramas de clases de las etapas 1, 2 y 3 a la documentación (-3 Puntos a Diagramas de Clases).
- 201530004-6 y 201530009-7 no realizan ningún commit, por ende, no se considera que hayan hecho uso del repositorio. 201403016-9 realiza demasiados commits con las etapas completadas, no entregando un avance progresivo y detallado del proyecto(-3 Puntos a Uso de Repositorio GIT).
- No se hace uso adecuado de JavaDoc al no utilizar rótulos para variables a retornar y porque no se genera página para la clase *BotoneraCabina.View* (-3 Puntos a Uso JavaDoc).
- En etapa 1, no se colocan un indicadores de piso en las botoneras mostradas (-2 Puntos a Etapa 1).
- En etapa 3, no se coloca un indicador de piso en la botonera de la cabina (-2 Puntos a Etapa 3).
- En etapa 3, la implementación del JSplitPane no permite aprovechar las ventajas que entrega este tipo de contenedor (-3 Puntos en Etapa 3).

XIV-B. Recomendaciones

- Tratar de no sobrecargar el readme de información, tratar de administrar lo informado entre este archivo y el pdf de documentación.

XIV-C. Aspectos Destacados

- El formato visual del readme es muy bueno, es agradable a la vista y permite una lectura sencilla.
- Agregar Hotkeys para acceder a los menús es un acierto que demuestra estudio sobre el sistema.
- El manejo de las excepciones es bastante interesante y refleja un buen entendimiento de la metodología.

XV. GRUPO 14 :201221075-5 - 201221031-3

XV-A. Descuentos

- las soluciones y dificultades se se proponen son repetitivas, es decir en una etapa dicen que ya tenían solucionado algo y en la siguiente etapa lo vuelven a mencionar lo cual no da mayor información de sus avances en el desarrollo iterativo e incremental (-2 Puntos Dificultades y Soluciones).
- falta la documentación de la clase interna *botoneraCabinaView* (-3 puntos Uso de JavaDoc).
- mala implementación de makefile (1 punto Descuentos varios).
- no se aprecia el número de donde se encuentra el ascensor etapa 1 (-2 puntos etapa 1).
- la etiqueta de los indicadores de donde está el ascensor se debe mostrar desde el principio y no recién cuando pase el piso 2 (-1 puntos etapa 2).
- en la etapa 3 no se considera apagar el botón de la cabina si apreto un botón del piso en cual estoy (-2 puntos Etapa 3).

XV-B. Recomendaciones

- cuando se presenten las soluciones que uds consideren las correctas, tienen que tener claro que todo debe funcionar en aragon que es donde los ayudantes probamos los códigos.

XV-C. Aspectos Destacados

- muy buen formato de informe.

XVI. GRUPO 15 : 201430003-4 - 201404583-2 - 201430001-8

XVI-A. Descuentos

- 201430003-4 genera muy pocos commits, los cuales no generan un avance significativo para el proyecto. 201430001-8 solo hace 1 commit y no se considera como un avance significativo (**-2 Puntos a Uso de Repositorio GIT**).
- No se hace uso adecuado de JavaDoc al no generar página para la clase *BotoneraCabina.View* (**-2 Puntos a Uso JavaDoc**).
- En etapa 3, no se hace implementación del JSplitPane (**-5 Puntos a Etapa 3**).

XVI-B. Recomendaciones

- El color de fondo de la cabina no es el más adecuado, ya que se podría confundir con que se encuentra en algún estado, tal como el sensor se vuelve rojo al detectar la cabina.
- El código generado para pintar la cabina y los sensores es redundante, dado que se puede hacer uso de los métodos heredados de la clase *Rectangle2D.Double* y un manejo adecuado de referencias.
- No crear atributos que no serán inicializados ni pedir argumentos que no serán utilizados.
- Eliminar comentarios tipo *fill in* al finalizar el código.
- Evitar hardcoding al realizar iteraciones.
- La clase motor no debiese revisar los datos proveniente de la interfaz gráfica, porque no se estaría cumpliendo la arquitectura Modelo Vista Controlador.

XVI-C. Aspectos Destacados

- El readme tiene un formato adecuado, además de entregar la información precisa y necesaria.
- El archivo de documentación se encuentra muy completo, entrega la información necesaria y además permite conocer de mejor manera como fue el desarrollo del programa gracias a la forma en que se plantearon las dificultades y soluciones.
- El uso de rótulos html permite que la documentación JavaDoc tenga más dinamismo.
- El código se encuentra comentado de una forma bastante detallada.

XVII. GRUPO 16 :201530008-9 - 201530001-1 - 201530002-K

XVII-A. Descuentos

- A pesar de las indicaciones previas, entregan archivos que no corresponden a la tarea, lo que pasa a ser archivos basura (**1 Punto en Descuentos Varios**).
- en la cuarta etapa todo funciona bien pero el cambio de velocidad no se ve reflejada en la transición del ascensor, lo cual no estaría correcto (**-3 Puntos en Etapa 4**).

XVII-B. Recomendaciones

- procuren de revisar todos los archivos que se suben a git para no caer con archivos basura.
- no generar commit que tengan la misma descripción ya que no genera información para quien revisa, por ahora solo sera un warning para la prox. se descontara.
- recuerden leer bien lo que se pide, ya que generaron la documentacion de toda la etapa 4 y solo era de botoneracabina y su vista de esta etapa, para que no tengan que hacer trabajo que no se evaluara.

XVII-C. Aspectos Destacados

- Excelente readme, claro y conciso.
- formato del informe impecable.

XVIII. GRUPO 17 : 201551010-5 - 201530024-0 - 201630015-5

XVIII-A. Descuentos

- No se incorporan diagramas de clases de las etapas 1, 2 y 3 a la documentación (-3 Puntos a Diagramas de Clases).
- Los commits son bastante informales, no reflejan totalmente el progreso del proyecto. Además, no se realiza el tag *Final*, por lo que no se respeta el formato de entrega de la tarea (-3 Puntos a Uso de Repositorio GIT).
- No se hace uso adecuado de JavaDoc al utilizar esporádicamente rótulos para parámetros y variables a retornar (-2 Puntos a Uso JavaDoc).
- El readme no detalla como ejecutar el programa, ni como usar efectivamente el archivo *makefile* (-3 Puntos a README).
- El formato de entrega de la documentación es poco ordenado, no incorpora una portada con los integrantes, no es agradable a la vista y las imágenes utilizadas contienen elementos que no corresponden a la ejecución del programa, lo que no permite establecer un enfoque claro en ellas (-1 Punto a Entrega de Documentación General).
- En etapa 1, no se colocan un indicadores de piso en las botoneras mostradas (-2 Puntos a Etapa 1).
- En etapa 2, el sensor no refleja adecuadamente el comportamiento que posee, se mantiene prendido un periodo después de que la cabina ha pasado y se enciende antes de que el ascensor llegue cuando la cabina baja (-5 Puntos a Etapa 2).
- En etapa 3, botones de la caja ascensor no reflejan si han sido presionados. Además no se coloca un indicador de piso en la botonera de la cabina (-4 Puntos a Etapa 3).
- En etapa 3, la implementación del JSplitPane no permite aprovechar las ventajas que entrega este tipo de contenedor (-3 Puntos a Etapa 3).
- En etapa 4, el menú no respeta el formato mostrado en las instrucciones. Además, en las modificaciones de velocidad y de tiempo de pausa se pueden ingresar valores negativos, lo que conlleva a un funcionamiento erróneo (-6 Puntos en Etapa 4).
- A pesar de las indicaciones previas, entregan archivos .class, lo que pasa a ser archivos basura (1 Punto en Descuentos Varios).

XVIII-B. Recomendaciones

- Es de suma importancia realizar el tag *Final*, dado que este permite fijar el estado de los archivos al realizar la entrega de la tarea. En futuras tareas el no cumplimiento de esto derivará en declarar a la tarea como no entregada.
- Explicar claramente como utilizar el archivo *makefile* en el readme.
- Mantener la formalidad al utilizar el repositorio GIT y entregar más detalles en los mensajes de los commits.
- No generar más carpetas si no se explica que contiene cada una de ellas.
- Generar la documentación de javadoc en una carpeta distinta a donde se almacenan los códigos.
- Al solicitar datos por ingresar, indicar claramente las unidades de medición a utilizar.
- Eliminar comentarios tipo *to be coded* al finalizar el código.
- Al realizar el pintado de los sensores, el cambio de color se hace dentro de la vista de caja ascensor. Se recomienda que esto se haga dentro de la vista del sensor en sí y que la caja del ascensor solo se encargue de llamar al dibujado del elemento.

XIX. GRUPO 18 :201130007-6 - 201130004-1

XIX-A. Descuentos

- A pesar de las indicaciones previas, entregan archivos que no corresponden a la tarea, lo que pasa a ser archivos basura (1 Punto en Descuentos Varios).
- se ha recalcado que debe ser un diagrama de clase para cada etapa de la tarea ya que así se puede apreciar de mejor manera los avances que tiene cada grupo en cada etapa (-2 Punto a Entrega de Documentación General).
- no se hace uso de javadoc (-6 Puntos a Uso JavaDoc).
- no se hace uso de la plataforma git de manera constante solo al final para la entrega, lo cual no refleja un uso adecuado de esta (-3 Puntos a Uso de Repositorio GIT).
- En etapa 3, la implementación del JSplitPane no permite aprovechar las ventajas que entrega este tipo de contenedor (-3 Puntos a Etapa 3).
- error de compilación etapa 4 (-15 Puntos a Etapa 4) (-10 Puntos a Sin errores y sin warnings).

XIX-B. Recomendaciones

- limpiar los códigos de comentarios que no tengan relación con la entrega de la tarea.

XX. GRUPO 19 :201103003-6

XX-A. Descuentos

- No se incorpora diagrama de clase de la etapa 4 a la documentación (-1 Puntos a Diagramas de Clases).
- Sólo se hace 1 commit, subiendo completo el proyecto, lo que va en contra de la idea de poseer un repositorio que almacena la historia y el progreso del trabajo (-3 Puntos a Uso de Repositorio GIT).
- No se hace uso de JavaDoc (-6 Puntos a Uso JavaDoc).
- El readme no explica como utilizar el archivo makefile, no entrega información relevante del programa y se divide en cada uno de las etapas, en vez de entregar una para la tarea en general (-3 Puntos a README).
- No se mencionan las dificultades presentadas durante el desarrollo de la tarea ni como se resolvieron (-6 Puntos a Dificultades y Soluciones)
- No se hace entrega de archivo de documentación (-4 Puntos a Entrega de Documentación General).
- En etapa 2, el sensor no refleja adecuadamente el comportamiento que posee, se mantiene hasta que otro sensor se activa, lo que resulta inexacto (-5 Puntos a Etapa 2).
- No se realiza la etapa 4 (-15 Puntos a Etapa 4).

XX-B. Recomendaciones

- Avisar con antelación cuando posea complicaciones en el trabajo con su grupo.
- Mantener formalidad en redacción del readme.
- Eliminar comentarios tipo *to be coded* al finalizar el código.
- Evitar hardcoding al realizar iteraciones.
- No agregar métodos ni atributos en las clases si no van a ser utilizados.
- No dejar código (métodos, atributos, clases, entre otros) no utilizado comentado en la entrega final de la tarea.
- Evitar switch muy extensos para asignar imágenes, se pueden dar nombres genéricos a la imágenes, de forma que permitan ser utilizados en una iteración, por ejemplo "1.png".

XXI. GRUPO 20 :201221024-0 201121047-6 - 201104136-4

XXI-A. Descuentos

- el uso de git fue después de la fecha final de entrega , pero hubo un buen uso de este, lo cual no puede tener un puntaje completo (-2 Puntos a Uso de Repositorio GIT).
- no hubo para nada documentación de las clases solo ejecutaron el comando javadoc sobre la clase pero no se explica nada en esta clase, no se les puede considerar el puntaje completo (-5 Puntos a Uso JavaDoc).
- En etapa 3, la implementación del JSplitPane no permite aprovechar las ventajas que entrega este tipo de contenedor (-3 Puntos a Etapa 3).
- fallas generales en la etapa 4 lo cual no permite la perfecta ejecucion del programa (Ej: el ascensor se va al infinito) (-10 Puntos etapa 4).

XXI-B. Recomendaciones

- para hacer javadoc no es necesario tener la clase compilada.
- no incluir información que ya tenían prevista en el readme esto conlleva a información innecesaria.

XXII. GRUPO 21 : 201790126-8 - 201321060-0**XXII-A. Descuentos**

- No se incorporan diagramas de clases de la etapas 1, 2 y 3 a la documentación (**-3 Puntos a Diagramas de Clases**).
- A pesar de las indicaciones previas, se entregan archivos readme por etapa, en vez de uno solo general, lo que pasa a generarse mucha información redundante (**1 Punto en Descuentos Varios**).

XXII-B. Recomendaciones

- El diagrama de clase no necesita mostrar los métodos y atributos, dado que termina por saturar la vista del diagrama.
- La documentación termina por estar saturada debido a que cerca de la mitad de ésta corresponde a la descripción del enunciado y de requerimientos. Tratar de resumir más estos aspectos.
- Eliminar comentarios tipo *to be coded* al finalizar el código.
- Evitar hardcoding al realizar iteraciones.
- No agregar métodos ni atributos en las clases si no van a ser utilizados.
- No dejar código (métodos, atributos, clases, entre otros) no utilizado comentado en la entrega final de la tarea.
- La clase *MyElevatorFrame* no debiese tener métodos relacionados con la manipulación de los datos ingresados por los usuarios, porque no se estaría cumpliendo la arquitectura Modelo Vista Controlador.
- La utilización de las excepciones es poco robusta, al igual que el tratamiento de el ingreso de datos negativos y caracteres no numericos.

XXII-C. Aspectos Destacados

- El código se encuentra comentado de una forma bastante detallada.