

# Desarrollo de una Interfaz Gráfica para una Herramienta de Cálculo de Estructuras

Rafael Olivera - Federico García

27 de noviembre de 2015



# Índice

<b>1. Introducción</b>	<b>6</b>
1.1. Definición del problema y motivación . . . . .	6
1.2. Desarrollo previo . . . . .	7
1.3. Objetivos y resultados esperados . . . . .	8
1.4. Desarrollo del proyecto . . . . .	9
1.5. Organización del documento . . . . .	9
<b>2. Estado del Arte</b>	<b>10</b>
2.1. Cálculo de estructuras . . . . .	10
2.1.1. Cálculo implicados . . . . .	10
2.1.2. IETFEM . . . . .	10
2.2. Herramientas comerciales . . . . .	10
2.2.1. SAP2000 . . . . .	11
2.2.2. AxisVM . . . . .	11
2.2.3. Herramientas Web . . . . .	11
2.3. Desarrollo 3D . . . . .	12
2.3.1. OpenGL . . . . .	12
2.3.2. Java 3D . . . . .	12
2.3.3. WebGL . . . . .	12
2.3.4. Otras herramientas . . . . .	12
2.4. Desarrollo 3D en la Web . . . . .	12
2.4.1. HTML5 - Canvas . . . . .	12
2.4.2. Librerías para desarrollo 3D . . . . .	12
2.4.3. Interacción con el usuario . . . . .	13
2.5. Información complementaria . . . . .	13
2.5.1. Investigación sobre proyectos similares en América Latina	13
2.5.2. Herramientas de cálculo de estructuras en la Web . . . . .	13
<b>3. Organización del trabajo</b>	<b>13</b>
3.1. Alcance . . . . .	13
3.2. Metodología de trabajo . . . . .	15
3.3. Estimación y esfuerzo efectivo . . . . .	16
<b>4. Presentación de la solución</b>	<b>18</b>
4.1. Análisis y relevamiento de requerimientos . . . . .	18
4.2. Diseño de la solución . . . . .	18
4.2.1. Decisiones tomadas . . . . .	18
4.2.2. Diseño final . . . . .	18
4.3. Arquitectura . . . . .	18
4.4. Tecnologías y herramientas utilizadas . . . . .	19
4.4.1. HTML5 - Javascript - CSS3 . . . . .	19

4.4.2.	Bootstrap . . . . .	19
4.4.3.	AngularJS . . . . .	19
4.4.4.	ThreeJS . . . . .	19
4.4.5.	Electron . . . . .	19
4.5.	Manejo del espacio 3D . . . . .	19
4.5.1.	Eventos de usuario . . . . .	19
4.5.2.	Adición, sustracción y transformación de objetos . . . . .	20
4.5.3.	Manejo de la cámara . . . . .	20
4.5.4.	Trazado de rayos e intersecciones con objetos . . . . .	20
4.5.5.	Performance . . . . .	20
4.6.	Manejo de datos . . . . .	21
4.6.1.	Entrada de información (dibujado e importación) . . . . .	21
4.6.2.	Mantenimiento de la estructura durante el proceso de dibujado . . . . .	21
4.6.3.	Almacenamiento de la estructura . . . . .	21
4.6.4.	Salida de Datos . . . . .	21
4.7.	Análisis de resultados del Core . . . . .	22
4.7.1.	Generación de resultados . . . . .	22
4.7.2.	Introducción de datos en la UI . . . . .	22
4.7.3.	Visualización . . . . .	22
<b>5.</b>	<b>Resultados obtenidos</b>	<b>22</b>
5.1.	Comparación IETFEEM con y sin UI . . . . .	22
5.1.1.	Análisis del impacto en la usabilidad . . . . .	22
5.1.2.	Análisis del impacto en el tiempo de ejecución . . . . .	22
5.2.	Casos de prueba . . . . .	23
5.2.1.	Estudio de casos de pequeño porte (Torre pequeña) . . . . .	23
5.2.2.	Estudio de casos de mediano porte (Grúa) . . . . .	23
5.2.3.	Estudio de casos de gran porte y pruebas de stress (Torre Eiffel) . . . . .	23
<b>6.</b>	<b>Conclusiones y trabajo futuro</b>	<b>23</b>
6.1.	Conclusiones . . . . .	23
6.2.	Trabajo a futuro . . . . .	24
6.2.1.	Trabajo en el motor . . . . .	24
6.2.2.	Trabajo en la interfaz . . . . .	24
6.2.3.	Despliegue de la aplicación . . . . .	24
<b>7.</b>	<b>Anexos</b>	<b>24</b>



# 1. Introducción

## 1.1. Definición del problema y motivación

Desde las primeras casas construidas por el hombre, hasta el edificio más moderno y extravagante que exista en la actualidad, puede decirse que se buscó en el fondo el mismo objetivo: lograr una estructura segura, resistente y funcional. Hoy por hoy, la evolución del conocimiento humano y de la tecnología circundante ha permitido desarrollar a niveles altísimos el entendimiento del problema y sus posibles soluciones.

El cálculo de estructuras, en ese sentido, es una rama fundamental dentro de la ingeniería civil. Se trata de una serie de complejos cálculos realizados con la finalidad de lograr estructuras óptimas con las condiciones descritas anteriormente. A grandes rasgos, se busca que la estructura pueda soportar tanto su propio peso, como cualquier fuerza externa que pueda ser aplicada a la misma, lo que puede derivar en que la misma no siempre sea igual en la realidad a como se diseñó. Esto quiere decir que la estructura puede sufrir ciertas deformaciones antes de alcanzar su punto de equilibrio.

La Ingeniería en Computación no ha dejado este problema de lado, ya que existen diversos sistemas informáticos encargados de facilitar el diseño y cálculo de estructuras. Estos sistemas permiten, a grandes rasgos, dibujar una estructura mediante la definición de diferentes materiales, secciones, fuerzas externas, etc. Finalmente, realizan los cálculos correspondientes, mostrando la estructura en un estado de equilibrio y las deformaciones ocurridas en el proceso.

Así como existen estos sistemas reconocidos mundialmente, la Facultad de Ingeniería cuenta también con su propio sistema de cálculo de estructuras. Su nombre es IETFEM, y fue desarrollado por los Ing. Pablo Castrillo y Jorge Pérez pertenecientes al Instituto de Estructuras y Transporte(IET). Se trata de un motor de cálculo desarrollado en Octave que recibe una estructura descrita en formato texto y genera gráficas e imágenes con la deformación de la misma.

En este proyecto, se realizará una interfaz gráfica acorde para ser utilizada en conjunto con el motor de cálculo antes mencionado, y lograr así un sistema completo de diseño y cálculo de estructuras. Se busca, en particular, agregar funciones de dibujo y visualización de resultados que pueden observarse en otros sistemas de la misma índole, acercando al IETFEM a los sistemas comerciales y logrando una mayor usabilidad y eficiencia para los estudiantes que lo utilizan.

## 36 1.2. Desarrollo previo

37 Como mencionamos anteriormente, la FING cuenta con un motor de cálculo  
38 de estructuras desarrollado por los Ingenieros Pablo Castrillo y Jorge Pérez. El  
39 mismo resuelve problemas de cálculo de estructuras utilizando el Método de  
40 Elementos Finitos(MEF).

41 El MEF es, desde mediados del siglo XX, una de las principales herramientas  
42 utilizadas por los ingenieros para el análisis de sistemas estructurales, mecáni-  
43 cos, eléctricos, etc. El avance de la computación y la disponibilidad creciente de  
44 computadores potentes a bajo costo ha provocado que los programas comercia-  
45 les de MEF para el cálculo estructural sean utilizados masivamente. De hecho,  
46 en los últimos cuarenta años el MEF ha transformado los procedimientos de tra-  
47 bajo de todas las áreas de ingeniería y constituye hoy una de las herramientas  
48 indispensables con las que un ingeniero debe contar en el ejercicio de su profe-  
49 sión. Por otra parte, el uso del MEF por parte de profesionales no debidamente  
50 capacitados podría eventualmente producir errores en el diseño de estructuras,  
51 y por tanto, riesgos para los usuarios.

52 En este contexto, la enseñanza del MEF en las carreras de Ingeniería se trans-  
53 forma en un desafío docente, donde además de formar a los estudiantes en el  
54 uso de diferentes programas de cálculo estructural es necesario transmitirles los  
55 conocimientos y herramientas que les permitan realizar un análisis crítico de los  
56 resultados. Es importante destacar además, que la mayoría de los programas  
57 comerciales (ej: SAP2000 y AxisVM) de MEF son de código cerrado, por lo que  
58 presentan como desventaja a nivel educativo, que no permiten a los estudiantes  
59 ver su funcionamiento interno, limitando la comprensión de los errores durante  
60 el aprendizaje.

61 De esta manera surge entre docentes del Grupo de Mecánica de Sólidos Comp-  
62 putacional (MSC) del Instituto de Estructuras y Transporte (IET) la motivación  
63 de brindar una solución al problema a través del desarrollo de un software edu-  
64 cativo adecuado: IETFEM.

65 IETFEM comenzó a desarrollarse en 2012 . El primer módulo desarrollado per-  
66 mitió resolver problemas de estructuras de barras articuladas o apoyadas en  
67 el plano con cargas aplicadas en los nodos. Esta primera versión fue utilizada  
68 por estudiantes del curso de Elasticidad 2013; luego se incluyó la posibilidad de  
69 generar un informe de salida en formato LaTeX. Posteriormente, la herramien-  
70 ta contó con el aporte del docente del IET, Agustín Spalvier, desarrollando la  
71 capacidad de ingresar cargas distribuidas uniformes en elementos de pórtico y  
72 el análisis modal de vibraciones de pórticos. Finalmente, a principios de 2014,  
73 Castrillo desarrolló un módulo para la resolución de problemas con variaciones  
74 de temperatura y fuerza de volumen en barras articuladas.

75 Se buscó una herramienta que sin ser compleja para su aplicación en cursos  
76 de grado, permita al estudiante visualizar el funcionamiento interno del méto-  
77 do de cálculo. Por ello se optó por la sintaxis de programación de GNUOctave

78 (herramienta libre de alta compatibilidad con Matlab), ya conocida por los es-  
79 tudiantes. Se considera que contar con un software abierto donde los estudiantes  
80 pueden entender e incluso programar nuevos cálculos de acuerdo a sus necesi-  
81 dades, enriquece el trabajo desde el punto de vista didáctico.

82 La forma de ingreso de datos se eligió de acuerdo a otros programas de cálculo  
83 de estructuras como SAP2000 donde se deben definir: materiales, secciones,  
84 estados de carga, geometrías, conectividades, etc. En el IETFEM se optó por  
85 una entrada de archivo de texto plano donde el estudiante debe ingresar esta  
86 información. La salida también es en texto plano (.txt y .tex) y gráfica, al igual  
87 que en los programas comerciales.

88 Sin embargo, la generación del archivo de entrada y la comunicación con el  
89 IETFEM pueden llegar a ser tediosas y complicadas para el estudiante. Debe  
90 tenerse en cuenta que debe especificarse la estructura nodo por nodo, barra  
91 por barra, describiendo los materiales, secciones y fuerzas aplicadas, entre otras  
92 cosas, respetando además un formato fijo de documento que puede derivar en  
93 diversos errores de sintaxis.

94 Por lo tanto, se desarrollará en este proyecto una interfaz gráfica de código  
95 abierto donde el estudiante pueda dibujar la estructura de una manera sencilla  
96 e intuitiva, y que genere la entrada al IETFEM de manera automática. De esta  
97 manera, se pretende mejorar tanto la facilidad de uso como la eficiencia del  
98 mismo.

### 99 **1.3. Objetivos y resultados esperados**

100 Como mencionamos antes, a pesar de la increíble potencia en la resolución del  
101 problema del cálculo de estructuras, IETFEM presenta ciertos puntos a mejorar  
102 para ser comparado con otros sistemas del mismo rubro.

103 A lo largo de este proyecto perseguimos 2 grandes objetivos que consideramos  
104 esenciales para el enriquecimiento del sistema: Mejorar la eficiencia y mejorar  
105 la usabilidad

106 Para mejorar la usabilidad, se desarrollará una interfaz que permita al usuario  
107 dibujar la estructura de manera fluida y amigable. Se trata de un espacio 3D  
108 donde el usuario puede moverse libremente utilizando el mouse para mover y  
109 rotar la cámara. Se podrá dibujar la estructura de una manera continua e in-  
110 intuitiva. Además, facilitará la comunicación con el motor de cálculo previamente  
111 desarrollado y la visualización de los resultados obtenidos.

112 Para mejorar la eficiencia, reduciremos el tiempo de ejecución del motor de  
113 cálculo, eliminando el proceso de graficación y generación de imágenes, ya que  
114 ahora los resultados podrán verse en la nueva interfaz. Como regla básica, bus-  
115 camos que el usuario pierda el menor tiempo posible en problemas tecnológicos  
116 o informáticos y que dirija sus esfuerzos al comprendimiento del problema en sí  
117 y su método de resolución.



118 A modo de resumen, se busca realizar un sistema ágil, de código abierto, que  
119 mejore ambos aspectos lo suficientemente como para poder ser utilizado sin pro-  
120 blemas en el curso de Elasticidad dictado por el IET. Con el fin de verificar el  
121 cumplimiento de los objetivos por parte del sistema, una vez finalizado, será  
122 evaluado resolviendo ejercicios del curso antes mencionado, realizando compa-  
123 raciones y análisis del tiempo de ejecución.

## 124 1.4. Desarrollo del proyecto

125 El proyecto comenzó con una fase fuerte de investigación. Inicialmente se reali-  
126 zaron reuniones ocasionales con los clientes, donde se reunió información valiosa  
127 sobre el problema de cálculo de estructuras y el método de elementos finitos.  
128 Además se definió qué tipo de herramienta se quería, qué funcionalidades eran  
129 deseadas y qué objetivos se buscaban. Durante esta etapa se utilizó el motor  
130 de cálculo directamente para comprender su funcionamiento y compararlo con  
131 otras herramientas comerciales.

132 Una vez comprendido el problema, se procedió a buscar herramientas con las  
133 cuáles desarrollar la interfaz. Se investigaron librerías y lenguajes de programación  
134 3D, optando al final por utilizar tecnologías web por su simplicidad de uso,  
135 agilidad y portabilidad

136 Posteriormente se comenzó a diseñar e implementar la herramienta, separan-  
137 do en diferentes módulos que serán descriptos en detalle en el capítulo 4. Se  
138 ejecutaron reuniones quincenales con los clientes para definir detalles, corregir  
139 errores, evaluar resultados y tomar decisiones en conjunto. Esta fase ocupó la  
140 mayor parte del tiempo del proyecto, debido a la dificultad técnica del mismo.

141 Finalmente, una vez alcanzado un producto inicial que cumplía las expectativas  
142 planteadas, se procedió a realizar pruebas sobre el mismo, detectando ciertos  
143 errores de performance que fueron solucionados hasta un nivel considerable-  
144 mente bueno(se hablará de estas medidas en el capítulo 5).

## 145 1.5. Organización del documento

146 El resto del documento se organiza de la siguiente manera:

147 En el siguiente capítulo comenzamos analizando el estado del arte, tanto del  
148 problema de cálculo de estructuras como de herramientas de programación 3D,  
149 y su posible uso en sistemas de este tipo. Se realiza un estudio de diferentes  
150 herramientas investigadas, el estado de las mismas y su posibilidad de ser uti-  
151 lizadas en este proyecto. También se investigan otros sistemas de cálculo de  
152 estructuras y otros proyectos académicos similares en América Latina.

153 Posteriormente, en el capítulo 3, hablaremos de la organización del trabajo a  
154 lo largo del proyecto. Hablaremos del alcance del mismo, definiendo las fun-

155 cionalidades y características específicas que se buscan en el producto final. Se  
156 describirá la metodología de trabajo utilizada y se realizarán estimaciones para  
157 cada tarea comprendida, comparando finalmente con el esfuerzo efectivo.

158 A continuación, en el capítulo 4, se procederá a plantear la solución propuesta,  
159 detallando cada aspecto de la misma. Se describirá con exactitud su proceso  
160 de diseño e implementación, la arquitectura definida, el funcionamiento de cada  
161 componente, las herramientas utilizadas y su uso en general.

162 En el capítulo 5, se especifican los resultados obtenidos, analizando diferentes  
163 casos de prueba y comparando con resultados obtenidos desde IETFEM antes  
164 de la realización de este proyecto. Se analizan además los problemas obtenidos  
165 durante esta fase y cómo fueron resueltos.

166 Finalmente, el 6 capítulo, enumera las conclusiones obtenidas durante el pro-  
167 yecto, analizando el cumplimiento de objetivos y proponiendo posible trabajo a  
168 futuro a desarrollar sobre IETFEM.

## 169 **2. Estado del Arte**

### 170 **2.1. Cálculo de estructuras**

#### 171 **2.1.1. Cálculo implicados**

172 aca hay que hablar el problea, les mandamos un mail paraver donde podemos  
173 leer mas

#### 174 **2.1.2. IETFEM**

175 hablar de como resuelve el ietfem los problemas de arriba  
176 beneficios del ietfem, resuelve probelmas complejos como los comerciales  
177 carencias del ietfem, mencionar lso problemas un pocomas en detalle, mas que  
178 nada haciendo referencia a que a pesar de que soluciona probleas complejos  
179 eficientemente, asi como esta es muy dificil de usar

### 180 **2.2. Herramientas comerciales**

181 Existen en el mercado diversos productos de software enfocados al análisis de  
182 estructuras, con gran cantidad de funcionalidades y utilizados por ingenieros de  
183 todo el mundo en problemas reales. En el marco de este proyecto se exploraron  
184 con mayor rigurosidad dos herramientas: SAP2000 y AxisVM, las cuales en  
185 etapas mas avanzadas del desarrollo fueron tomadas como estándar para la  
186 implementación de ciertas funcionalidades.

### 187 **2.2.1. SAP2000**

188 Es un software comercial desarrollado por la empresa Computers & Structures, Inc. fundada en 1975 en California, siendo uno de los pioneros en herramientas de análisis de estructuras.

191 Actualmente en su versión 18, SAP2000 es una aplicación para computadoras que se ejecuta en ambientes Windows. Cuenta con un entorno gráfico 3D para el modelado y una interfaz de usuario muy completa que puede resultar demasiado compleja en los primeros pasos.

195 Entre las características mas importantes se encuentran:

- 196 ■ Un motor de análisis que puede resolver varios tipos de problemas tales como: (LES PREGUNTAMOS A LOS PROFES XQ ES UN HUEVO DE ENTEDNER),
- 199 ■ Diversas características para el modelado como templates, sistema de grillas, distintas vistas y herramientas de meshing.
- 201 ■ Diversos componentes estructurales como articulaciones, barras, cables sólidos, resortes, etc.
- 203 ■ Posibilidad de aplicar distintos tipos de cargas.
- 204 ■ Varias posibilidades para ver la salida de los cálculos con diagramas, tablas y reportes.
- 206 ■ Importación y exportación de modelos en distintos formatos estándar.

207 Por todo esto SAP2000 es uno de los productos líderes en el mercado siendo utilizado en mas de 160 países en todo el mundo.

### 209 **2.2.2. AxisVM**

210 Es un software comercial desarrollado por la empresa InterCAD Kft. en 1991 y con sede en Hungría. Fue una de las primeras herramientas 3D basada en el método de los elementos finitos.

213 Actualmente en su versión 13, AxisVM requiere computadoras con el sistema operativo Windows. Cuenta con un entorno gráfico 3D para el modelado y una interfaz de usuario muy completa que puede resultar demasiado compleja en los primeros pasos.

### 217 **2.2.3. Herramientas Web**

218 hablar de cada uno, quien lo desarrolla, si se puede en que esta hecho

219 hablar del impacto mundial, que problemas resuelve, hablar de las cosas que nos gustaria poner en nuestro programa, cual nos gustaria evitar o mejorar

## 221 **2.3. Desarrollo 3D**

### 222 **2.3.1. OpenGL**

223 investigar y hablar de open gl, que es, en que se usa, etc...

224 hablar de porque no la elegimos, dificultad de uso, poca experiencia, etc...

### 225 **2.3.2. Java 3D**

226 lo mismo que al anterior

### 227 **2.3.3. WebGL**

228 lo mismo que el anterior

229 hablar ademas de que al descubrir esta opcion se nos desperto la idea de hacerlo  
230 eb

### 231 **2.3.4. Otras herramientas**

232 hablar de otras herramientas de escritorio que hayamos visto, no se que mas  
233 poner aca

## 234 **2.4. Desarrollo 3D en la Web**

### 235 **2.4.1. HTML5 - Canvas**

236 que es html5

237 Hablar de las facilidades que da el html5 para cosas 3d mediante el canvas

238 posibles conexiones entre ebgl y canvas

### 239 **2.4.2. Librerías para desarrollo 3D**

240 hablar de las libreirias que vimos sobre ebgl, threejs y las otras que estuvios  
241 viendo que no me acuerdo

242 porque elegimos threejs y que beneficios se tienen

243 hacer un analisis si se pueden cumplir los objetivos con esta tecnologia

### 244 **2.4.3. Interacción con el usuario**

245 como pretendemos que sea la interaccion con el usuario, basandonos en los  
246 programas comerciales y lo que ofrece la eb  
247 plantearse cambiar el titulo de esta seccion para abarcar mas contenido

## 248 **2.5. Información complementaria**

### 249 **2.5.1. Investigación sobre proyectos similares en América Latina**

250 buscar proyectos similares en internet, en america latina y el mundo, y compa-  
251 rarlos  
252 comparar lo encontrado con ietfem, y rematar señalando que es el primer pro-  
253 yecto de este tipo en sudamerica

### 254 **2.5.2. Herramientas de cálculo de estructuras en la Web**

255 Investigar si existen  
256 en casod e que existan, hacer una mini comparacion con lo que seria ietfem eb  
257 mencionar que no existan muchas herramientas y que tendria mucho potencial

## 258 **3. Organización del trabajo**

### 259 **3.1. Alcance**

260 Como mencionamos anteriormente, los objetivos planteadas en este proyecto se  
261 basan en mejorar tanto la eficiencia como la usabilidad del IETFEM. En ese  
262 sentido, existen dentro de la rama del cálculo de estructuras una infinidad de  
263 funcionalidades y mejoras posibles que pueden resultar útiles en nuestro sistema.  
264 Por lo tanto, se definió un conjunto acotado de funcionalidades y característi-  
265 cas deseables en el producto final, apuntando a alcanzar satisfactoriamente los  
266 objetivos planteados y lograr una herramienta de alto nivel.

267 Se consideró como prioridad apuntar a una herramienta académica, es decir, una  
268 herramienta libre, intuitiva para los estudiantes y aplicable en cursos dictados  
269 por el IET. En particular, se tomó como referencia el curso de Elasticidad, curso  
270 donde ya fué utilizado satisfactoriamente el IETFEM y donde será utilizado  
271 luego de la realización de este proyecto.

272 La principal y más grande funcionalidad que se desarrollará será le presencia  
273 de un espacio 3D. El mismo será el elemento central de la aplicación, mediante

la cual el usuario efectuará la mayor parte de las interacciones posibles. Se pretenden integrar dentro de este espacio 3D las siguientes funcionalidades:

- Rotación de la cámara de visualización.
- Movimiento de la misma por todo el espacio 3D.
- Zoom In y Zoom Out.
- Dibujado de nodos y barras.
- Dibujado de grillas auxiliares.
- Selección de nodos y barras: Para setear propiedades en las mismas.
- Eliminación de nodos y barras.
- Visualización de propiedades: Fuerzas, puntos de apoyo y resortes.
- Visualización de estructura resultante: Observar la deformada y comparar con estructura original.
- Escalamiento la estructura deformada: «Exagerar» la deformación, para apreciar pequeñas deformaciones.
- Visualización de las propiedades de la estructura deformada utilizando escalas de colores: Deformación, Fuerzas, Tensiones, etc.

Más allá de que se pretende que el usuario tenga una experiencia interactiva mediante el dibujo de la estructura, es necesario definir ciertas funcionalidades fuera del espacio 3D. Ya sea tanto por comodidad como por intuición, estas opciones se encuentran en diferentes menús que rodean el espacio, similar a los demás programas comerciales dentro del rubro que se investigaron en el capítulo anterior.

- Abrir y Guardar Estructuras.
- Definición de Materiales.
- Definición de Secciones.
- Asignar propiedades a barras: Material y sección.
- Asignar propiedades a nodos: Fuerzas, Apoyos y Resortes.
- Selección de nodos y barras: Para setear propiedades en las mismas.
- Eliminación de nodos y barras.
- Prendido y apagado de elementos auxiliares
- Seteo de Factor de escalamiento para la estructura deformada

Si bien estos elementos nos permiten estimar una interfaz gráfica completa e intuitiva, resta definir aún la funcionalidad más importante del proyecto: la comunicación con el motor de cálculo. La salida de la interfaz debe ser un archivo reconocible por el IETFEM, del cuál pueda obtener todos los datos de

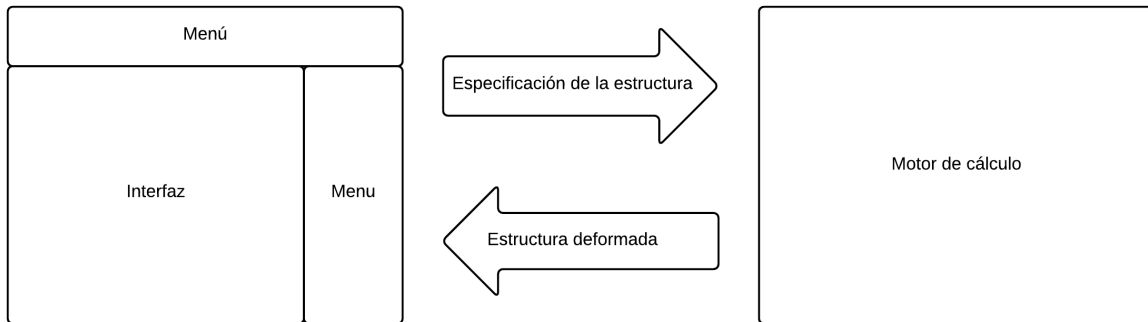


Figura 1: Ciclo de vida de IETFEM

la estructura. Así mismo, el motor debe ofrecer como salida otro archivo, el cuál será recibido por la interfaz con el fin de mostrar los resultados obtenidos. Dicha comunicación puede observarse en la Figura 1, aunque se hará hincapié en cómo se resolvió esta comunicación en el siguiente capítulo.

### 3.2. Metodología de trabajo

En las primeras etapas del proyecto se focalizó el trabajo en comprender el problema que se quiere resolver. Se tuvieron reuniones quincenales con los clientes dónde se habló del problema del cálculo de estructuras y cómo lo resuelve IETFEM. Dichas reuniones se apoyaron además en una permanente comunicación por e-Mail y una vasta investigación del problema por nuestra parte. Para esto no sólo se investigó sobre el problema, sino que además se utilizaron productos similares e incluso el propio IETFEM con ejemplos simples.

Una vez comprendido el problema, se pasó a buscar una solución al mismo. Dentro de esta etapa se pueden incluir la búsqueda de herramientas, el análisis y el diseño de la aplicación. Se mantuvieron las reuniones con los clientes, evaluando herramientas y enseñando prototipos realizados a modo de prueba. Se investigaron lenguajes y librerías de programación 3D, tanto de escritorio como web, decidiendo en última instancia utilizar WebGL (se hablará más en detalle en el siguiente capítulo).

Conforme pasaba el tiempo las reuniones se fueron enfocando cada vez más en el producto final, comenzando a definir las funcionalidades y características del mismo. Mientras se mantenía contacto con los clientes, se realizó por nuestra parte la definición de casos de uso, con sus respectivos diagramas de flujo, al tiempo que se definió la arquitectura del sistema en base a los requerimientos obtenidos y las prestaciones de las herramientas definidas.

Finalmente, para las etapas de implementación y testing, se creó un repositorio

Descripción	Estado	Prioridad
Agregar la posibilidad de definir un material por defecto al dibujar barras	Resuelta	1
Agregar sprites y flechas a los nodos cuando se definen fuerzas o condiciones de desplazamiento	Resuelta	1
Agregar funcionalidad para "prender" y "apagar" grillas y fuerzas	Resuelta	2
Agregar funcionalidad Abrir Modelo / Guardar Modelo	En proceso	2
Agregar funcionalidad Nuevo Modelo, que limpia la escena y permite definir las nuevas unidades de medida	Para hacer	3

Figura 2: Planilla Excel utilizando metodología Kanban

en Github con el esqueleto de la aplicación y todo código reusable proveniente de la etapa de prototipación. Como metodología de trabajo se utilizó la metodología ágil Kanban. Kanban es un método para gestionar el trabajo intelectual, con énfasis en la entrega justo a tiempo, mientras no se sobrecargan a los miembros del equipo. En este enfoque, el proceso, desde la definición de una tarea hasta su entrega al cliente, se muestra para que los participantes lo vean y los miembros del equipo tomen el trabajo de una cola.

Existen diversas herramientas on-line de planificación y gestión de proyectos, tales como Jira o TFS. Sin embargo, debido a la poca cantidad de personas involucradas en el proyecto (2 desarrolladores y 2 clientes) y a que las tareas a realizar estaban bien definidas, se optó por utilizar una herramienta simple y natural: una planilla Excel online. La misma se encontró en todo momento de libre acceso y modificación para los 4 participantes, y cada tarea tenía asignada una descripción, un estado, y una prioridad.

En un principio, se agregaron todas las tareas a realizar, y ambos desarrolladores tomaban cada una de ellas marcando su estado como «En proceso». Una vez finalizada, se marcaba la tarea como terminada y se subían los cambios al repositorio, marcando cada subida con la funcionalidad correspondiente.

A su vez, los clientes, los cuáles también tenían acceso a la última versión del IETFEM, iban relevando en la misma planilla problemas o cosas a mejorar que se encontraban en el producto, los cuáles pasaban a ser parte de nuestra «pizarra de kanban» y seguían el mismo flujo que las demás tareas.

### 3.3. Estimación y esfuerzo efectivo

La planificación del tiempo se realizó tomando en cuenta el desconocimiento inicial del problema de cálculo de estructuras y la dificultad de la programación gráfica en 3D. En la Figura 3 se pueden ver las estimaciones realizadas calculando 15 horas de trabajo semanal por desarrollador. Se puede apreciar que el período de trabajo se calculó entre Abril y Diciembre, logrando un total de 34 semanas de trabajo, que se traducen en un total de 1020 horas de trabajo.

Podemos ver también que ciertas etapas se planificaron en simultáneo por ciertos períodos de tiempo, especialmente en las etapas tempranas del proyecto



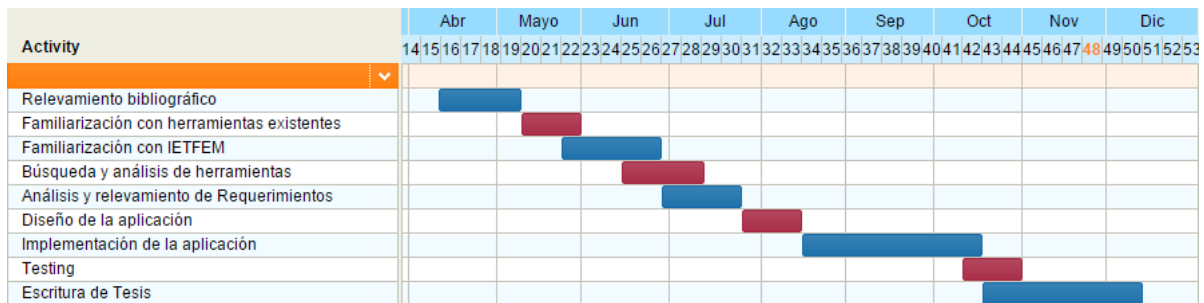


Figura 3: Diagrama de Gantt con la planificación del proyecto

366 donde se comenzó utilizando y comprendiendo tanto el IETFEM como otras  
367 herramientas, mientras se iba definiendo al mismo tiempo cómo realizar la inter-  
368 faz. Se planificó de esta manera debido a que se consideró que sería bueno  
369 evaluar varias herramientas en simultáneo, a modo de comparar y definir qué  
370 funcionalidades y características nos gustaría que estén presentes en nuestro sis-  
371 tema. También evaluar cómo llevarlas a cabo utilizando las herramientas que  
372 existen en el mercado y el contexto académico en el cuál se quiere insertar la  
373 aplicacación.

374 También se observa la concurrencia de tareas en los instantes finales del desa-  
375 rrollo, donde se planificó al mismo tiempo el testing y la escritura de la Tesis.  
376 Debido a la metodología ágil elegida y al tiempo estipulado, resulta conveniente  
377 que el testeo de la aplicación comience cuanto antes, ya que corregir un error  
378 pasará a ser parte de nuestra cola de tareas, y dependiendo de la prioridad de  
379 la misma podría ser resuelta antes que otras tareas definidas anteriormente pe-  
380 ro con una baja prioridad. La escritura de la tesis se planificó en simultáneo  
381 simplemente para intentar reducir el tiempo total del proyecto.

382 El cronograma estimado se realizó de manera exitosa, siguiendo cada etapa en el  
383 orden estipulado sin demoras excesivas. Como agregado, durante la implemen-  
384 tación se descubrieron nuevas funcionalidades que serían útiles en el sistema, las  
385 cuales fueron evaluadas con los clientes y algunas de ellas se llevaron a cabo sin  
386 problemas, debido a que la metodología de trabajo lo permitía.

387 También es necesario destacar el tiempo invertido en la Ingeniería de Muestra  
388 a fines del mes de Octubre, el cuál contempló el diseño de carteles, presentación  
389 del proyecto y la propia presencia en el evento. Esto redujo unos días el tiempo  
390 estipulado para la escritura de la tesis, el cuál se intenta recuperar durante  
391 el mes de noviembre aumentando la cantidad de horas a un promedio de 20  
392 semanales por desrrollador dedicadas a dicha tarea.

## 393 4. Presentación de la solución

### 394 4.1. Análisis y relevamiento de requerimientos

395 hablar de las reuniones que tuvimos en la que explicaron el problema y fuimos  
396 deduciendo los casos de uso, mencionarlos

397 hacer una lista de requerimientos

398 referenciar los casos de uso en el anexo

### 399 4.2. Diseño de la solución

#### 400 4.2.1. Decisiones tomadas

401 HABLAR DE PORQUE LO HICIMOS EB Y LOCAL

402 hablar de las reuniones interminables sobre el servidor y sus preocupaciones por  
403 el mantenimiento a futuro y otras complicaciones

404 hablar brevemente de electron como posible alternativa

405 hablar de las decisiones de no hacer grandes deformaciones, de hacer solo reti-  
406 culados de los atributos que no tomamos en cuenta, etc

#### 407 4.2.2. Diseño final

408 hablar de que dadas las condiciones decidimos hacerlo eb pero manteniendo todo  
409 estatico, para que despues no haya problemas de compatibilidad

410 mostrar un diagrama de diseño

411 hablar de cada modulo

### 412 4.3. Arquitectura

413 mostrar la arquitectura mediante diagramas de distribucion de componentes y  
414 fisicas

415 mostrar ademas como quedaria en version servidor

416 distribucion de componentes: hablar de cada componente: manejo del espacio,  
417 del modelo, edicion de puntos, de lineas etc

## 418    **4.4.    Tecnologías y herramientas utilizadas**

### 419    **4.4.1.    HTML5 - Javascript - CSS3**

420    dedicar un parrafo a cada una y como lo usamos particularmente en nuestro  
421    proyecto

### 422    **4.4.2.    Bootstrap**

423    que es y como lo utilizamos en nuestro proyecto  
424    que mejoras se tienen con respecto a no utilizarlo

### 425    **4.4.3.    AngularJS**

426    lo mismo que arriba

### 427    **4.4.4.    ThreeJS**

428    mencionar que es la libreria principal  
429    como funciona, habar que etsa sobre ebgl, que funcione en un canvas, que se  
430    programa mediante javascript, etc  
431    enque lo usamos en el proyecto  
432    mencionar y mostrar otros proyectos con three

### 433    **4.4.5.    Electron**

434    mencionr que nosparecio esencial para la prolijidad del proyecto "local", ya que  
435    si aspiramos a que el estudiante quelo usa no tenga que lidiar con cosas de  
436    cimpuacion, seria contraproducente que tenga que abrir un html pelado en el  
437    navegador, donde puede tener problemas de compatibilidad con navegadores  
438    diferentes, etc

## 439    **4.5.    Manejo del espacio 3D**

### 440    **4.5.1.    Eventos de usuario**

441    como definimoslo que puede hacer el usuairo  
442    los eventos definidos para el mouse y latecla de escape

443 que componentes se comunican directamente con las acciones de usuario y des-  
444 cribir .el camino"que realia cada una de ellas

#### 445 **4.5.2. Adición, sustracción y transformación de objetos**

446 hablar de la escena, como se agregan o modifican objetos

447 como se conforma un objeto, atributos relevantes

448 hablar del renderizado

449 acciones que se pueden hacer desde el programa

#### 450 **4.5.3. Manejo de la cámara**

451 hablar de orbit controls

452 hablar de elcambio de ejes de coordenadas para que el queda arriba

453 hablar como se mueve la camara tratar de ver el js asqueroso y deducir mate-  
454 maticamente como se mueve con respecto al espacio 3d para dibujarlo o mostrar  
455 un diagramita

#### 456 **4.5.4. Trazado de rayos e intersecciones con objetos**

457 hablar de como resolvimos la intersección del 'click' con el rayo de la camara para  
458 seleccionar objetos

459 hacer formulacion matematica

460 hablar de las cosas que three automatia y de como quedo resuelto

461 que compontes se enargan de esto

#### 462 **4.5.5. Performance**

463 hablar de los probelmas que encontramos al probar la torre eiffel

464 describir que se investigo y se hicieron camnios

465 hablar de los geometry de los objetos

466 de sacar a seleccion al hacer hover

467 de otras performances que se hicieron

468 hablar finalmente de otras mejoras que sepodrian hacer pero resultarian innece-  
469 sarios porque serian muy complejos par este proyecto en el que los estudiantes  
470 nunca vana hacer una estructura tan grande

## 471 **4.6. Manejo de datos**

472 hablar de como es la estructura que guardamos

### 473 **4.6.1. Entrada de información (dibujado e importación)**

474 que informacion guardamos de cada elemento y porque

475 en que momento agregamos cosas al mdoelo

476 hablar del dibujado

477 hablar de la importacion(como reemplamos el modelo)

478 hablar de abrir y guardar proyectos(como reemplamos el modelo)

### 479 **4.6.2. Mantenimiento de la estructura durante el proceso de dibu-** 480 **jado**

481 aca hablamos de como manipulamos los objetos, modificar y eliminae

482 hablar de mantener la consistencia, al abrir un modelo nuevo, guardar, abrir,

483 etc...

### 484 **4.6.3. Almacenamiento de la estructura**

485 hablar de que el modelo se va guardando en una variable javascript

486 hacer un estudio sobre que tan eficiente es y si la memoria del navegador

487 'da'para almacenar algo asi

488 mostrar un mini ejemplo y exactamente qué se guarda

### 489 **4.6.4. Salida de Datos**

490 hablar de lo que se genera desde la ui

491 como se genera, proceso que ace el usuairo para generarlo

492 validaciones que se toman en cuenta

493 como manipulamos el modelo ara generar el txt

## 494   **4.7.   Análisis de resultados del Core**

### 495   **4.7.1.   Generación de resultados**

496   hablar sobre qué genera el core

497   mencionar las cosas que agrega el texto

### 498   **4.7.2.   Introducción de datos en la UI**

499   como se ingresan

500   como se modelan y almacenan esos datos - deformedmodel

### 501   **4.7.3.   Visualización**

502   qué se ve

503   hablar de las opciones que se tienen

504   como hicimos el sitcheo entre deformada en indeformada

505   como hicimos el escalamiento

506   como hicimos el colorado

## 507   **5.   Resultados obtenidos**

### 508   **5.1.   Comparación IETFEM con y sin UI**

#### 509   **5.1.1.   Análisis del impacto en la usabilidad**

510   hablar de opinion de estudiantes, posiblemente en la idm

511   mostrar unto por punto en que aspectos se mejoraron

#### 512   **5.1.2.   Análisis del impacto en el tiempo de ejecución**

513   usar ietfem viejo y nuevo y calcular tiempos

514   ver si es muy dificil hacer un mini servidor para hacer una comparaciond de  
515   tiempos mejor

## 516 **5.2. Casos de prueba**

### 517 **5.2.1. Estudio de casos de pequeño porte (Torre pequeña)**

518 hablar de que comenamos con ese  
519 se utilio para realiar la primera integracion con el core  
520 sedescurbrieron probelmasde ejes y se resolvieron facilmente

### 521 **5.2.2. Estudio de casos de mediano porte (Grúa)**

522 se intento realiar el caso inicialmente como una prueba de stress  
523 no se encontraron problemas  
524 cuando se constato que funcionaba bien, se decidio utiliar un caso mas grande  
525 utiliado en la idm para mostrar le funcionamiento  
526 hablar de que ya se considera exitoso que funcione bien para la grua ya que lso  
527 estudiantes nunca van a hacer algo tan grande

### 528 **5.2.3. Estudio de casos de gran porte y pruebas de stress (Torre** 529 **Eiffel)**

530 comentar que se decidio hacer latorre eiffel para ver como respondia el programa  
531 hablar del trabajo de la perfomrmance y memoria  
532 cuanto se mejoro luego de los arreglos  
533 importancia de que ande 'perfecto'ya que es un caso inalcanable

## 534 **6. Conclusiones y trabajo futuro**

### 535 **6.1. Conclusiones**

536 hablar si las estimaciones y el esfuerxo fueron acertados  
537 si se cumplieron los objetivos  
538 evaluar la herramienta

## 539 **6.2. Trabajo a futuro**

### 540 **6.2.1. Trabajo en el motor**

541 que se puede agregar en el motor

542 porticos

543 osibilidad de migrar a otro lenguaje e integrar en un solo proyecto con la ui

### 544 **6.2.2. Trabajo en la interfaz**

545 agregar cosas que ya se pueden hacer en el core

546 mejoras de performance

547 otros 'chiches' que tienen programar comerciales

548 delegar responsabilidades a otra aplicación

### 549 **6.2.3. Despliegue de la aplicación**

550 hablar del servidor

551 como funcionaria con servidor y porque no se hizo así

552 donde se podría alojar

553 mejoras que implicar en el sistema

## 554 **7. Anexos**

555 mini Manual de uso

556 ejemplos de estructuras

557 modelo de dominio

558 casos de uso

559 diagramas de flujo

560 diagramas de arquitectura

561 mas info threejs

562 mas info otros proyectos similares

563 otras cosas XD