

Proyecto de grado

Informe Final

Siniestros en Locales de Gran Aglomeración de Público



Integrantes:

Laura Cuadrado
Germán Roldán
Gonzalo Segovia
Marcelo Sosa

Tutores:

Omar Viera
Sandro Moscatelli

Departamento de Investigación Operativa
Instituto de Computación
Facultad de Ingeniería
Universidad de la República

aglomeraciones

Mayo 2010

RESUMEN

En los últimos tiempos el desarrollo urbano se ha caracterizado por las grandes aglomeraciones de público en centros comerciales, eventos musicales, deportivos, etc. La emergencia de un siniestro, en un evento o lugar de tal naturaleza, amerita contar con planes de respuesta en lo que hace a la operativa de contención, atención, evacuación de la población involucrada y de heridos, accesibilidad de vehículos de emergencia y mitigación de impactos en el entorno urbano inmediato. Las respuestas deben coordinar múltiples actores en tiempos cortos, por lo que es necesario contar con información relevante para la toma de decisiones al momento del evento.

El presente proyecto tiene como objetivo el desarrollo de una herramienta de software que contribuya eficazmente a la prevención y mitigación de siniestros en locales de gran aglomeración de público. Para ello, se realizó una investigación de los antecedentes, actualidad y perspectivas de futuro sobre la gestión de riesgo general, y, en particular, sobre siniestros en locales de gran aglomeración de público.

Como resultado de la investigación realizada, surgió la necesidad de que el software ofrezca información relevante tanto, para la prevención de un siniestro así como, para minimizar el impacto en caso de que ocurra.

El software finalmente construido, representa un prototipo del sistema, que resuelve el problema planteado a satisfacción del usuario.

En este documento, inicialmente se presentan los requerimientos que debe tener el software para cumplir con las necesidades y expectativas del cliente. Posteriormente se muestra el método de solución, diseño e implementación de un prototipo de acuerdo a las necesidades planteadas, así como las pruebas realizadas. Por último, se presentan las conclusiones obtenidas tras desarrollar el sistema descrito, y sugerencias acerca de trabajos futuros.

Palabras clave: riesgo, siniestro, aglomeraciones, grafo, evacuación.

Índice

ÍNDICE	4
1 INTRODUCCIÓN	6
1.1 Antecedentes.....	6
1.2 Definición del problema	8
1.3 Alcance.....	8
1.4 Método Implementado	9
1.5 Resultados.....	10
1.6 Conclusiones.....	11
1.7 Organización del Documento.....	12
2 DESCRIPCIÓN DEL PROBLEMA	14
2.1 Definiciones	14
2.2 Descripción General del Sistema.....	15
2.3 Alcance.....	16
2.4 Definición del Alcance	16
3 REQUERIMIENTOS	18
3.1 Local	18
3.2 Plano.....	18
3.3 Piso	18
3.4 Personas	18
3.5 Siniestro.....	19
3.6 Salidas del sistema.....	19
3.7 Simulación	19
3.8 Usabilidad.....	19
3.9 Generalización.....	20
3.10 Riesgo cero.....	20
4 MÉTODO DE SOLUCIÓN	21
4.1 La Simulación	21
4.2 Algoritmos empleados.....	22
5 DISEÑO	41
5.1 Consideraciones generales de diseño.....	41
5.2 Arquitectura.....	41
5.3 Diagrama de Clases.....	44
6 IMPLEMENTACIÓN	47
6.1 Diagrama de Implementación	47
6.2 Tecnologías Empleadas.....	48
7 TESTEO	50
7.1 Descripción de los test realizados.....	50
7.2 Resultados Obtenidos	51
8 CONCLUSIONES Y TRABAJOS FUTUROS	52
8.1 Resultados alcanzados	52
8.2 Conclusiones.....	56

8.3	Trabajos futuros.....	59
9	MANUAL DE USUARIO.....	61
9.1	Página de Inicio	61
9.2	Módulo Diseño	62
9.3	Módulo Simulación	69
10	REFERENCIAS BIBLIOGRÁFICAS.....	80
11	ANEXOS.....	83



1 Introducción

En esta sección se da una introducción a la temática del proyecto, enfocado en los siniestros en locales de gran aglomeración de público.

En los últimos tiempos el desarrollo urbano se ha caracterizado por las grandes aglomeraciones de público en centros comerciales, eventos musicales, deportivos, etc. En un evento de tales características la ocurrencia de un siniestro puede ser caótica si no se cuentan con planes de emergencia y contingencia.

Al ocurrir un siniestro, por la importante población flotante que se congrega en un evento o lugar, amerita contar con planes de respuesta en tiempos breves, en lo que respecta a la evacuación de los heridos y población en general congregada allí, así como también, con planes para la mitigación del siniestro. Para poder construir dichos planes se necesita de información relevante tanto para la prevención del siniestro como así también, para minimizar los daños en caso de que ocurra la emergencia.

Más allá de los objetivos específicos que nos planteamos, cabe mencionar que el objetivo principal del proyecto, consiste en proveer de conocimientos y herramientas que ayuden a preservar y salvar vidas humanas.

1.1 Antecedentes

A modo de tomar conciencia sobre la importancia del tema, se estudió antecedentes de siniestros ocurridos en eventos y locales con gran aglomeración de público, tanto en Uruguay como en otros países. A continuación se describen algunos de ellos:

2006 - Tragedia del Tren en Young – 8 muertos, mas de 20 heridos

La prueba para el programa "Desafío al Corazón", que se emitía por Canal 10, consistió en que un grupo de personas moviera una locomotora y dos vagones, tirando de unas sogas.

Más de un centenar de vecinos de Young realizaba la maniobra frente a la estación local, en un clima festivo del que participaban todos sus habitantes, con banderas y globos de colores, ya que el festival era para recaudar fondos para su hospital.

La demostración se realizó luego de una lluvia y varios de los que estaban más cerca de la locomotora resbalaron en el barro y cayeron, mientras que los restantes siguieron tirando, con lo cual la máquina continuó su lento avance y arrolló a unas 20 personas.

Cuando la cinchada se detuvo, por los parlantes se pidió despejar el lugar, mientras camilleros y vecinos ayudaban a retirar heridos de bajo la formación. En ese momento se comprobó que había varios muertos [1].

2004 - Centro Comercial Ycuá Bolaños en Asunción, Paraguay - Incendio - 396 muertos

El incendio del supermercado Ycuá Bolaños fue una de las más grandes tragedias sucedidas en Asunción, Paraguay.

El domingo 1 de agosto de 2004 se produjeron dos grandes explosiones en el interior del recinto comercial que albergaba, entre varias cosas, un supermercado, un estacionamiento para vehículos, oficinas comerciales y un restaurante. En las cocinas de este último lugar ocurrió una explosión que provocó un gigantesco incendio.

Según informes policiales, al momento de producirse el accidente, los guardias de seguridad cerraron las puertas del recinto para evitar que las personas, que colmaban el centro comercial, escaparan con bienes que no hubieran pagado. Como resultado 396 personas murieron (la mitad de ellas niños) y dejando a más de 500 con heridas de diversa consideración, incluyendo a 56 personas con quemaduras de tercer grado.

La ocurrencia de varios de estos siniestros alertaron a la población y llevaron a que los gobiernos municipales y nacionales tomaran medidas de prevención, principalmente desde el punto de vista legal. También dieron lugar a variadas organizaciones e instituciones de plano nacional, regional o internacional, abocadas a la prevención y mitigación de desastres [2].

2004 - Discoteca Cromañón de Buenos Aires, Argentina - Incendio - 194 muertos, 700 heridos

República Cromañón era una discoteca ubicada en la zona de Once en la ciudad de Buenos Aires, Argentina, trágicamente conocida a raíz de un incendio que comenzó la noche del 30 de diciembre de 2004, durante un recital de la banda de rock Callejeros. Este incendio provocó una de las mayores tragedias no naturales en Argentina, causando la muerte de 194 personas y dejando más de 700 heridos.

El incendio provocó también importantes cambios políticos y culturales. En relación a lo político, la Legislatura de la Ciudad de Buenos Aires inició un juicio político para destituir al entonces Jefe de Gobierno Aníbal Ibarra, por considerarlo responsable político de la tragedia [3].

1.2 Definición del problema

Como se mencionó anteriormente, la ocurrencia de siniestros en ámbitos de gran aglomeración pública puede tener consecuencias desastrosas, a nivel material y principalmente humano, si no se cuenta con medidas preventivas con respuestas rápidas y efectivas. Los cálculos y procesamientos computacionales aportan grandes ventajas en ese sentido. Es así que surge la necesidad de generar una aplicación que contribuya a prevenir y optimizar el proceso de respuesta ante eventos adversos, los cuales, pueden ir desde un incendio parcial en una zona determinada a varios focos o derrumbes que bloqueen una o varias salidas o puertas internas.

Las medidas de prevención pueden surgir a partir de simulaciones mediante el estudio de zonas vulnerables, los puntos críticos del local en evaluación, o las rutas óptimas de evacuación, obtenidas tras la simulación.

Cuando nos referimos a aglomeración de público, existen dos situaciones claramente diferenciables:

- Las aglomeraciones en ámbitos físicos concretos como centros comerciales, donde existen disposiciones de regulación para minimizar la posibilidad de accidentes.
- Las aglomeraciones efímeras, asociadas a convocatorias más o menos preestablecidas como ser la "Noche de las Luces", conciertos musicales o los actos políticos. Las mismas están condicionadas al lugar donde se realizan, no siendo lo mismo por ejemplo, si se realiza en un lugar abierto como ser la rambla o si es un concierto en un estadio con determinada y limitada cantidad de salidas.

Así mismo, en ambas situaciones se debe tener en cuenta por ejemplo: lugar, día, hora, cantidad y tipo de público según: horarios, feriados, días fríos, etc.

Es vital destacar que en las aglomeraciones en locales cerrados, es mayor el riesgo por tener a la gente contenida dentro del mismo y en general, existen pocas zonas de acceso y/o salida.

1.3 Alcance

Tras la especificación de requerimientos (ver Documento de Requerimientos, en capítulo Anexos) obtenida a partir de reuniones e información proporcionada por el cliente, se pretende que se desarrolle un prototipo enfocado en un ejemplo concreto (aglomeraciones en lugares cerrados), y se establezcan criterios que permitan su generalización.

Esta generalización refiere a otro tipo de locales permanentes, así como también a establecer las consideraciones que permitan extenderlo a grandes eventos

públicos como los reseñados anteriormente (ver Documento de Análisis, en capítulo Anexos).

Evacuación del local

La evacuación del local implica entre otras acciones, determinar las vías de evacuación de personas, para ello se debe tomar en cuenta: lugar donde se realiza el evento, área que abarca, cantidad y características de las personas involucradas, acceso de equipos de emergencia, etc.

También se debe considerar para la evacuación, los diferentes sectores funcionales del local, los metros cuadrados ocupados y las puertas de salida con las que se cuenta y las implicancias que esto tiene en cuanto a la velocidad y vías de evacuación.

Por último, las salidas esperadas de la aplicación están dirigidas principalmente al personal encargado de definir estrategias de prevención y mitigación de siniestros, como es el caso del Centro Coordinador de Emergencias Departamentales de Montevideo (CECOED) [4]. Las salidas incluyen la determinación de zonas vulnerables, puntos críticos, rutas óptimas de evacuación y rutas de acceso para los equipos de emergencia.

1.4 Método Implementado

El prototipo construido abarca, la etapa de diseño del ambiente a evaluar (construcción del local), y la simulación de un siniestro sobre los locales diseñados.

Para la etapa de construcción del local, se utiliza el concepto de matrices, donde cada celda de la matriz representa un espacio físico de un piso del plano.

La evacuación a partir de un siniestro determinado en un piso del local, el acceso al siniestro, la determinación de rutas óptimas, zonas vulnerables y puntos críticos, son los elementos claves del sistema implementado. Es por esto que una vez construido el plano, se procede a la generación de un grafo que abarque y contenga todos los elementos presentes en el plano (paredes, puertas, salidas, zonas intransitables, personas, siniestros). Sobre dicho grafo, se ejecutan, algoritmos tales como Dijkstra[5], Floyd[6] y Búsqueda de puntos críticos[6], adaptados a la realidad del proyecto y descriptos con más detalle en el capítulo 4 "Método de solución".

Luego de aplicado el algoritmo correspondiente se visualizan los resultados en la matriz que representa al plano en simulación.

1.5 Resultados

A continuación se enumeran los aspectos positivos y negativos del proyecto realizado.

Por negativos se hace referencia a aquellos requerimientos, que en acuerdo con el cliente (CECOED) y los tutores del proyecto fueron dejados fuera del alcance del proyecto.

Como se mencionó, aquí simplemente se enumeran los resultados alcanzados, en el capítulo 8 sección 1, se detallan con profundidad cada uno de estos.

Cuando hablamos de resultados positivos del proyecto, entendemos práctico dividir en resultados referidos a la investigación del proyecto y por otro lado los resultados alcanzados en el prototipo implementado.

Sobre la investigación podemos destacar:

- Extensa documentación (conocimiento) generada para la facultad y para el CECOED en el área Riesgos y Aglomeraciones.
- Aplicación de los conocimientos aprendidos durante la carrera, con el aditivo de haber sido aplicado en un proyecto real.
- Trabajar en un tema que no fue visto a lo largo de la carrera, que es transversal a varias disciplinas.
- La importancia de ser la primera investigación de gran porte en esta área en nuestro país, según palabras del director del CECOED.

Con respecto al prototipo, podemos agrupar los resultados positivos en cuatro grandes áreas: Tecnologías, Interoperabilidad, Extensibilidad y Facilidad de uso.

Tecnologías:

- Servidor único centralizado.
- Tecnologías de última generación.
- Software libre.

Interoperabilidad:

- Plataforma web.
- Funcionalidad para Importar/Exportar.

Extensibilidad:

- Del sistema construido.

- A futuro.

Facilidad de uso:

- Diseño del ambiente.

En acuerdo con el cliente (CECOED) y los tutores del proyecto, algunos requerimientos o funcionalidades, fueron dejadas fuera del alcance final de la implementación del prototipo, a continuación se describen aquellos resultados “negativos” del mismo.

Sistema:

- Sin manejo de Usuarios.

Diseño:

- Interface de construcción del piso.
- Locales cerrados.

Simulación

- Variedad de Salidas.

1.6 Conclusiones

La siguiente es una versión reducida de las conclusiones obtenidas, la versión completa se encuentra en el capítulo 8, sección 2.

Desde fines del siglo pasado, el desarrollo urbano ha tenido un gran incremento, a tal punto que se podría catalogar como exponencial.

En las últimas dos décadas han sucedido varios desastres o situaciones de emergencia en lugares o eventos con grandes aglomeraciones de público.

Lamentablemente, en general la discusión y cuestionamiento de estos temas se realiza después que suceden los desastres. Por ello, es fundamental cambiar esta mentalidad para comenzar a trabajar en la prevención del problema.

Las aglomeraciones conllevan riesgos, y el interés por su gestión ha aumentado, lo que se refleja en la creación de variadas instituciones y organizaciones, tanto nacionales como regionales e internacionales, dedicadas a la prevención y mitigación de los mismos.

En la evacuación del lugar concreto o local, los cálculos matemáticos y el manejo de grafos tienen un papel importante, ya que son necesarios para optimizar la evacuación tomando en cuenta la distribución de las personas, sus

velocidades, movimientos, las distancias hacia las salidas y el acceso a la zona siniestrada por parte del personal de emergencia.

Gracias al procesamiento computacional se obtienen datos valiosos en poco tiempo, realizando cálculos que a veces requieren mucho tiempo y dedicación obtenerlos por otros medios. De esta manera, los recursos informáticos podrían encargarse de procesar y realizar cálculos complejos, permitiendo a los especialistas en gestión de riesgo, "ganar" tiempo para definir estrategias a partir de los resultados obtenidos.

Creemos que el sistema desarrollado es un aporte valioso a la sociedad entera y nos enorgullece poder contribuir de alguna forma a la prevención de desastre y mitigación de riesgos, y sobre todo, la preservación de la vida humana .

La gestión de riesgos en general, y en particular, la de los siniestros en locales de grandes aglomeraciones de público, son temas muy amplios, que involucran una gran cantidad de factores; siempre hay cosas para mejorar.

El trabajo realizado pretende ser un avance más en el tema. No es una versión final, sino que intenta ser un aporte y motivador para seguir trabajando en el área, mejorándolo o iniciando nuevos estudios a partir de él.

Sin menospreciar lo ambicioso del objetivo principal del proyecto, la preservación de las vidas humanas, consideramos, que el proyecto desarrollado contribuye en cierta forma a lograr dicho objetivo.

1.7 Organización del Documento

Este documento ("informe final del proyecto"), tiene como objetivo: introducir los motivos del proyecto, el método de solución: describiendo las etapas de diseño, implementación y testeo del sistema, y brindar conclusiones acerca del trabajo realizado, así como también, describir posibles futuras extensiones.

El documento (dividido en once capítulos), se compone de la siguiente forma:

- Capítulo 1, Introducción:

En este capítulo, se brinda una introducción al proyecto, describiendo los antecedentes que motivaron el mismo, la definición y alcance del proyecto.

Una breve descripción del método implementado, los resultados, y las conclusiones obtenidas, tras la implementación del prototipo.

- Capítulo 2 Descripción del problema:

Aquí se dan algunas definiciones sobre las que se basa el resto del documento y se describe el problema tratado en el proyecto.

- Capítulo 3 Requerimientos:

Se detalla en profundidad, los requerimientos funcionales y no funcionales que conforman el prototipo desarrollado.

- Capítulo 4 Método de solución:

Se desarrollan y profundizan los algoritmos y las variantes aplicadas a los mismos, utilizados para la construcción del prototipo, y sus respectivas salidas.

- Capítulo 5 Diseño:

En este capítulo, se muestra el diseño del prototipo implementado, a través del diagrama de alto nivel de la arquitectura y el diagrama de clases.

- Capítulo 6 Implementación:

En esta sección, se detalla el diagrama de deployment utilizado en el prototipo implementado, así como las tecnologías seleccionadas para la construcción del mismo.

- Capítulo 7 Testeo:

Este capítulo, documenta las pruebas funcionales y no funcionales realizadas así como los resultados obtenidos a partir de las mismas.

- Capítulo 8 Conclusiones y trabajo futuro:

Como última parte del proyecto, se elaboró un conjunto de conclusiones, y posibles trabajos futuros a modo de expansión hacia el sistema final.

- Capítulo 9 Manual de usuario:

Este capítulo, contiene el manual de uso del prototipo.

- Capítulo 10 Referencias:

En esta sección del informe final, se encuentran las referencias a los documentos externos al informe, utilizados como apoyo para la elaboración del mismo.

- Capítulo 11 Anexos:

Adjunto al informe final del proyecto, se anexan, los documentos previos realizados. En este capítulo se enumera, el contenido de cada uno de los anexos.

2 Descripción del problema

En este capítulo se dan algunas definiciones sobre las que se basa el resto del documento y seguidamente se describe el problema tratado en el proyecto.

2.1 Definiciones

Aglomeración de público:

Toda reunión de un número plural de personas con propósitos lícitos, que se presente en cualquier edificación, instalación o espacio perteneciente a personas públicas o privadas, naturales o jurídicas o de uso público, que reúna las características cuantitativas y cualitativas que en las disposiciones pertinentes se indican, relacionadas con el número, la frecuencia, el lugar y las finalidades [7].

Emergencia:

Estado caracterizado por la alteración o interrupción intensa y grave de las condiciones normales de funcionamiento u operación de una comunidad, causada por un evento o por la inminencia del mismo, que requiere de una reacción inmediata y que exige la atención o preocupación de las instituciones del Estado, los medios de comunicación y de la comunidad en general [8].

Plan de contingencia:

Procedimientos operativos específicos y preestablecidos de coordinación, alerta, movilización y respuesta ante la manifestación o la inminencia de un fenómeno peligroso particular para el cual se tienen escenarios definidos [8].

Riesgo Público:

Se define como el daño probable que a causa o con ocasión de las aglomeraciones de público, se pueda producir sobre las personas y sus bienes, la infraestructura, la economía pública y privada y sobre el ambiente [7].

Siniestro:

- Conjunto de daños o perjuicios derivados de una misma causa.
- Avería grave, destrucción fortuita o pérdida importante que sufren las personas o las cosas por causa de un accidente, catástrofe, etc [9].

Pseudocódigo:

El pseudocódigo describe un algoritmo utilizando una mezcla de frases en lenguaje común, instrucciones de programación y palabras clave que definen las estructuras básicas. Su objetivo es permitir que el programador se centre en los aspectos lógicos de la solución a un problema.

2.2 Descripción General del Sistema

La aglomeración de público en espectáculos, centros comerciales, etc. es uno de los aspectos que caracterizan el desarrollo urbano en las últimas décadas. Las particularidades urbanas que han adquirido impactan en gran medida en los sistemas y estructuras territoriales, en particular los aspectos relacionados con infraestructuras y accesibilidades.

La emergencia de un siniestro en un evento de gran aglomeración de público amerita que se cuente con planes de respuesta referidos a la contención del siniestro, evacuación de la población, atención de los heridos y accesibilidad de los vehículos de emergencia. Por esto, es necesario que se cuente con información relevante para la toma de decisiones antes y durante el momento del siniestro debido a que en tiempos breves se deben coordinar múltiples actores.

El objetivo principal del proyecto es desarrollar un software que optimice el proceso de respuesta ante la ocurrencia de un siniestro en ámbitos de gran aglomeración pública. El siniestro puede ser, un incendio parcial en una zona determinada o varios focos a la vez que pueden ir bloqueando las salidas, por haber colapsado parcial o totalmente el techo.

Las salidas de la aplicación desarrollada pueden ser por ejemplo, una salida hacia el público en general sobre vías de escape y evacuación; otra hacia personal de seguridad del local, como zonas críticas, o vías de acceso al siniestro.

Se diferencian los siguientes ámbitos de aplicación:

- Las aglomeraciones en lugares físicos concretos como centros comerciales, donde existen disposiciones de regulación para minimizar la posibilidad de accidentes.
- Las aglomeraciones efímeras, asociadas a convocatorias más o menos preestablecidas como ser la "Noche de las luces", conciertos musicales y los actos políticos. Las cuales están condicionadas por ejemplo, al lugar donde se realizan (lugares abiertos como la rambla o cerrados como un concierto en el estadio con determinada y limitada cantidad de puertas).

En los ámbitos de aplicación descritos anteriormente se debe tener en cuenta por ejemplo: lugar, día, hora, cantidad y tipo de público según: horarios, feriados, días fríos, etc.

En aglomeraciones en locales cerrados, es mayor el riesgo por existir gente contenida dentro del mismo y, en general, existir pocas zonas de acceso y/o salidas.

2.3 Alcance

Como se menciona en la descripción general del sistema existen dos situaciones diferenciables, a continuación se detalla con más profundidad cada una de ellas.

2.3.1 Acceso y traslado de evacuados

Determinación de vías de acceso de unidades de emergencia y evacuación de heridos, según tipo de herido, tipo de centro asistencial, capacidades de los centros asistenciales, intensidad de tránsito en zonas adyacentes, dirección de las calles, eventualidad que una o varias de las vías de acceso estén bloqueadas.

Determinación de rutas óptimas y rutas óptimas seguras (que pasen lo menos alejado posible de puntos seguros).

2.3.2 Evacuación del local

La evacuación del local implica establecer criterios para determinar las vías de evacuación de personas (dependiendo del lugar donde se realiza, área que abarca, cantidad y características de las personas involucradas) y el acceso de equipos de emergencia.

Para dicha evacuación se debe considerar los diferentes sectores funcionales del local, los metros cuadrados ocupados y las puertas de salida con las que se cuenta, y las implicancias que esto tiene en cuanto a la velocidad de evacuación.

2.4 Definición del Alcance

Tras los estudios realizados, plasmados en el documento Estado del Arte Aglomeraciones (ver capítulo Anexos), y los siguientes argumentos provistos por los tutores, el destinatario del proyecto y nuestra investigación:

- El objetivo principal del proyecto es la preservación de las vidas humanas, en tal sentido, se considera que las acciones inmediatas en la zona siniestrada, son más críticas.
- Son varias las herramientas ya existentes y accesibles para hallar la ruta óptima entre 2 puntos de una ciudad, por ejemplo en Montevideo

(dispositivos GPS, sitios web que ofrecen dicha funcionalidad, tales como www.mapred.com o www.DUCSA.com.uy)

- En Uruguay, según datos proporcionados por el Centro Coordinador de Emergencias Departamentales, los centros comerciales no disponen de herramientas informáticas que den una respuesta rápida frente eventuales siniestros.
- En lo práctico se asume que el conductor de los vehículos de emergencia (ambulancia, bomberos, etc) sabrá llegar al punto del siniestro (Shopping, estadio) sin guiarse por un software en el momento de la emergencia.
- El hecho de calcular rutas óptimas es más aprovechable para la evacuación de las personas dentro de la zona siniestrada, dado que las mismas no necesariamente tienen conocimiento de las salidas de emergencia, ni de las vías por donde es más fácil transitar.
- Resulta más completo y útil el estudio dentro del local dado que en éste influyen directamente factores como: edad, sexo, características del local, tipo de siniestro, distribución y cantidad de población presente, etc.
- Luego de determinar la cantidad de personas que se evacuará por cada salida resultará más útil asignar los destinos de los evacuados.

En base a lo mencionado, y considerando también, las cotas en cuanto tiempo que debe tener el proyecto de grado, se acordó con el destinatario del proyecto y los tutores del mismo que, el alcance del proyecto, se enfoque en la evacuación de un local, dejando como trabajo futuro el acceso de unidades de emergencia y traslado de evacuados del área siniestrada.

Cabe mencionar que el objetivo es generar un sistema a modo de prototipo, que sea el inicio y motivador para la creación aplicaciones más completas y mejoradas.

Las salidas esperadas del prototipo están dirigidas principalmente al personal encargado de definir estrategias de prevención y mitigación de siniestros, como es el caso del CECOED. Las salidas incluyen la determinación de zonas vulnerables, puntos críticos, rutas óptimas de evacuación y rutas de acceso del equipo de emergencia.

3 Requerimientos

A continuación se detallan los requerimientos funcionales y no funcionales relevados a partir de reuniones con el CECOED, usuario principal del prototipo a construir (ver Documento de Requerimientos, en capítulo Anexos).

3.1 Local

El sistema debe manejar el concepto de "local", por un lado para mantener información básica acerca de cada local que puede ser modelado en el sistema, así como también para, establecer una relación entre el local, los planos asociados al mismo, las personas existentes en él, y los siniestros que pueden ocurrir.

3.2 Plano

Se pretende que el sistema permita el manejo de planos (asociados a un local), los mismos se corresponden a un área definida por el usuario, sobre la cual se agregan los distintos componentes, que modelan un local.

Dentro de los componentes que modelan el local se encuentran por ejemplo, los pisos y las conexiones entre ellos, como ser rampas, ascensores y escaleras.

El sistema debe permitir la persistencia de dichos planos (guardar, cargar, modificar y eliminar).

Por último el sistema debe ser capaz de validar el plano construido, por ejemplo, que entre dos pisos consecutivos, exista una vía (escalera o ascensor) que los comunique, que no existan áreas del plano inaccesibles, etc.

3.3 Piso

El sistema debe permitir el manejo de pisos asociados a un plano, sobre los cuales se podrían construir paredes, puertas internas, salidas y zonas intransitables. El piso con número cero indica la planta baja, y los números negativos, los subsuelos. Se debe controlar que los pisos sean consecutivos, es decir, que si existe el número 3 debe existir el 2, 1 y 0.

El sistema debe permitir la persistencia de dichos pisos.

3.4 Personas

El sistema debe de ser capaz de manejar los conceptos relacionados con las personas que se encuentran dentro de un plano (también asociadas al local).

En base a esto, el sistema debe de ser capaz de agregar personas al plano.

Debe de ser posible manejar la persistencia del concepto persona, es decir, permitir, guardar, modificar, cargar y eliminar personas de un plano.

3.5 Siniestro

Dado un plano correspondiente a un local, y un conjunto de personas de diversas características ubicadas en el mismo, el sistema debe de ser capaz de definir "siniestros" para su eventual simulación.

Al igual que para los planos y personas, debe ser posible agregar, modificar y persistir siniestros.

3.6 Salidas del sistema

Ya con todos los elementos disponibles (estructura del local, personas, y siniestro), el usuario, debe de seleccionar la o las salidas que desea obtener del sistema.

Dentro de las salidas deseables se encuentran:

- Calcular rutas óptimas de evacuación:
 - Para cada uno de los grupos de personas, determinar la ruta óptima de evacuación, de acuerdo a las características del Local y el siniestro producido.
- Calcular ruta óptima hacia el siniestro:
 - Para cada una de las Entradas/Salidas del Local, obtener la ruta o acceso óptimo hacia el Local, destinada a los equipos de emergencia o bomberos.
- Determinar nivel de vulnerabilidad de las áreas:
 - Para cada uno de los sectores del Local, determinar basado en la distancia a las Entradas/Salidas, y ubicación y características del siniestro, la vulnerabilidad de dichas zonas.
- Determinar puntos críticos:
 - Dado un Local, indicar, los puntos más vulnerables del Local, a modo de ejemplo, aquellas puertas internas de un piso, que de bloquearse, dejan a una zona del Local, sin acceso al exterior.

3.7 Simulación

El sistema debe de permitir la simulación del siniestro, para generar las salidas solicitadas o indicadas por el usuario.

3.8 Usabilidad

Se pretende que el sistema a construir sea:

- Robusto.
- Fácil de utilizar
- Eficiente.

3.9 Generalización

Pese a que se desarrolle el caso referido a un ejemplo concreto se deberían establecer criterios que permitan su generalización para todo tipo de locales, así como establecer las consideraciones que permitan extenderlo a grandes eventos públicos reseñados anteriormente.

3.10 Riesgo cero

Este proyecto de grado al estar enmarcado en un contexto de lugares con aglomeración de público, tiene como objetivo primordial, la prevención de la vida humana. De acuerdo a esto, cada método de solución que se emplee o algoritmo para construir el sistema, debe de manejar el concepto de "riesgo cero" como fundamental e ineludible. El concepto de "riesgo cero" significa, en el contexto del proyecto, que no existe ningún riesgo de vida para las personas en los lugares de aglomeración de público (al ocurrir un siniestro), o en su defecto que el riesgo se ha reducido al máximo.



aglomeraciones

4 Método de solución

A continuación se describe los algoritmos desarrollados en el prototipo de aglomeraciones y se brinda el pseudocódigo de los mismos.

4.1 La Simulación

El usuario selecciona la acción de ejecutar, el sistema, a partir del plano, la distribución de los grupos de personas, y las salidas especificadas a obtener, genera la simulación y cálculos correspondientes para elaborar los informes y salidas indicadas.

Ruta óptima de evacuación

El usuario selecciona como salida del sistema, el cálculo de las rutas óptimas de evacuación, el sistema en base al plano, la distribución de personas y características del siniestro, calcula las rutas óptimas de evacuación y genera la siguiente salida:

- Trayecto óptimo de evacuación para cada uno de los grupos de personas, hacia cada una de las salidas.

Ruta óptima hacia el siniestro

El usuario selecciona como salida del sistema, el cálculo de las rutas óptimas hacia el o las áreas de siniestro, el sistema en base al plano y la ubicación de los focos del siniestro, calcula desde cada una de las salidas, la vía óptima de acceso.

Determinar Puntos críticos

El usuario selecciona como salida del sistema la opción de determinar puntos críticos (puntos que dejan desconexo al grafo), el sistema calcula aquellas áreas que son puntos críticos en el plano, es decir aquellas áreas, que de no existir, o verse bloqueadas por algún siniestro, resultan críticas para la evacuación de personas, o acceso al siniestro de personal de emergencia. A partir del cálculo, el sistema elabora un informe, con las áreas o puntos críticos del plano, a modo de ejemplo, si la única vía de comunicación entre un piso y otro es la escalera, claramente la misma es un punto crítico del plano.

Niveles de vulnerabilidad

El usuario selecciona como salida del sistema, el cálculo de áreas vulnerables, el sistema en base al plano, sus características, y un siniestro, elabora un informe presentando, para cada área del plano el grado de vulnerabilidad existente, es decir aquellas áreas, que de suceder un siniestro, serán las más afectadas.

4.2 Algoritmos empleados

En esta sección se dará una breve descripción de los algoritmos usados para resolver las salidas del sistema y para la generación del grafo (nodos y aristas) a partir de las personas, zonas transitables y salidas.

4.2.1 Construcción de grafo a partir de matrices de celdas y conexiones

El grafo a construir representará los recorridos posibles dentro del local, considerando las vías transitables en cada piso y las conexiones (rampas, escaleras, ascensores). Las conexiones se podrán habilitar o deshabilitar, lo que es realizado principalmente para los ascensores, ya que, al tratarse de sistemas eléctricos, no es conveniente su utilización en caso de siniestros. Cada piso del local se representa como una matriz de celdas de tamaño fijo, las cuales definen un área determinada del piso (por ejemplo, de 0,5 por 0,5 metros). Esta es el área mínima representable, o sea que no se podrá especificar un elemento dentro del plano (por ejemplo, una pared) con un área menor al de una celda. El usuario especifica la escala (tamaño de las celdas) al momento de crear un plano. Cada celda pertenece a una posición de la matriz, determinada por la fila y columna.

Para aplicar algoritmos que permitan hallar las rutas óptimas, puntos críticos y niveles de vulnerabilidad, es necesario construir un grafo a partir del plano del local, o sea a partir de las matrices de los pisos y sus conexiones. Es así que, para cada matriz, se generará un subgrafo dentro del grafo total, y las conexiones (rampas, escaleras) se corresponderán con aristas que unen esos subgrafos.

En la figura 1 se muestra una matriz "vacía" de 20 filas por 30 columnas.

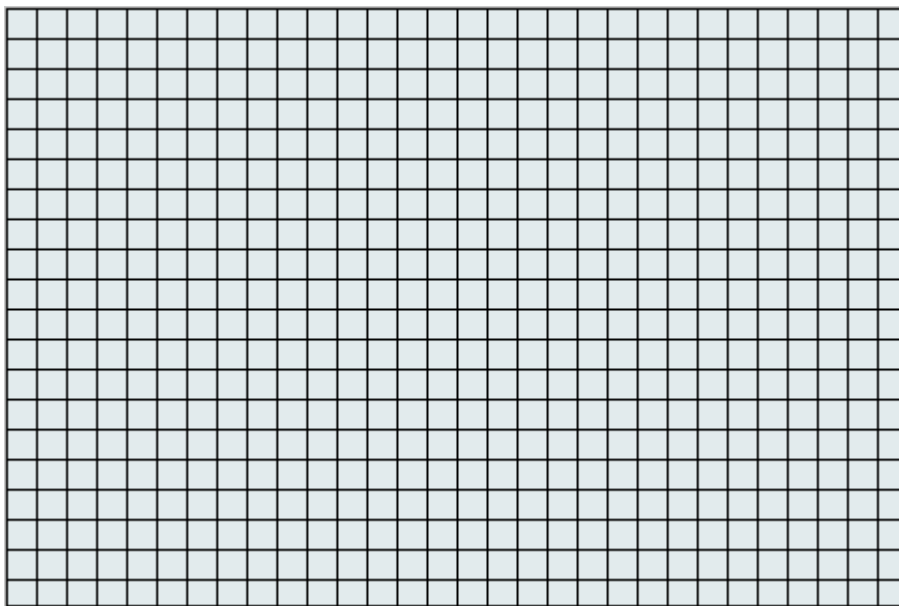


Figura 1 Matriz de Celdas que representa un piso

Cada celda tiene básicamente las siguientes propiedades:

- **Tipo:** indica lo que se encuentra en dicha posición o área del plano. Los valores posibles son:
 - TIPO_VACIA: el área delimitada es una zona transitable.
 - TIPO_PARED: el área delimitada es una pared.
 - TIPO_SALIDA: el área delimitada es una puerta de salida del local.
 - TIPO_PUERTA_INTERNA: el área delimitada es una puerta interna.
 - TIPO_ZONA_INTR: el área delimitada es una zona intransitable.
 - TIPO_ESCALERA: el área delimitada es el acceso a una escalera.
 - TIPO_RAMPA: el área delimitada es el acceso a una rampa.
 - TIPO_ASCENSOR: el área delimitada es el acceso a un ascensor.
 - TIPO_NODO: el área delimitada es una zona transitable que se corresponde con un nodo del grafo.
 - TIPO_ARISTA: el área delimitada es una zona transitable que se corresponde con una arista del grafo, o sea es una vía de evacuación.
- **idNodo:** identificador del nodo del grafo, si la celda es de tipo TIPO_NODO.
- **idArista:** identificador del nodo del grafo, si la celda es de tipo TIPO_ARISTA.

Las celdas de tipo TIPO_VACIA, TIPO_PARED, TIPO_SALIDA, TIPO_PUERTA_INTERNA, TIPO_ZONA_INTR, TIPO_ESCALERA, TIPO_RAMPA y TIPO_ASCENSOR son determinadas de acuerdo al plano construido por el usuario. Las celdas de tipo TIPO_NODO y TIPO_ARISTA se marcan internamente cuando se genera el grafo. Cada nodo o arista son un conjunto de celdas contiguas, las cuales tienen el mismo identificador (idNodo o idArista respectivamente).

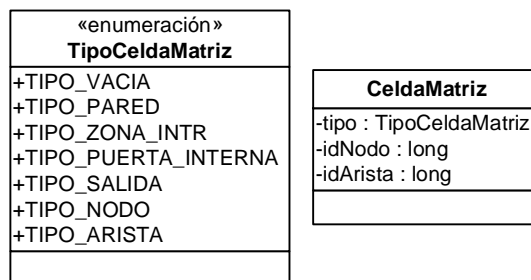


Figura 2 Celda Matriz

Por otro lado, el grafo a generar contendrá un conjunto de nodos y aristas. Como se mencionó anteriormente, en la matriz, los nodos son un conjunto de celdas contiguas que forman un rectángulo. Es así, que cada nodo se representa

por la ubicación (fila, columna) de la celda superior izquierda del mismo y el tamaño (alto, ancho).

Los nodos y aristas pueden ser de distintos tipos.

NODOS:

- **Persona:** zona del plano en el que se encuentran las personas inicialmente (nodo inicial).
- **Puerta Interna:** puerta interna del local.
- **Interno:** zona del plano desde la cual puede cambiar la dirección del recorrido hacia las salidas (evacuación) o hacia el siniestro (acceso del personal emergencia).
- **Conector:** punto de acceso a una conexión (escalera, rampa o ascensor) hacia otro piso.
- **Salida:** puerta de salida del local (nodo final).
- **Siniestro:** zona del plano en el que ocurre un siniestro.

ARISTAS:

- **Vía:** zona transitable que conecta dos nodos dentro de un piso.
- **Escalera:** conecta dos nodos de tipo conector (escalera).
- **Rampa:** conecta dos nodos de tipo conector (rampa).
- **Ascensor:** conecta dos nodos de tipo conector (ascensor).

aglomeraciones

La representación del grafo mediante un diagrama de clases se muestra en la figura 3.

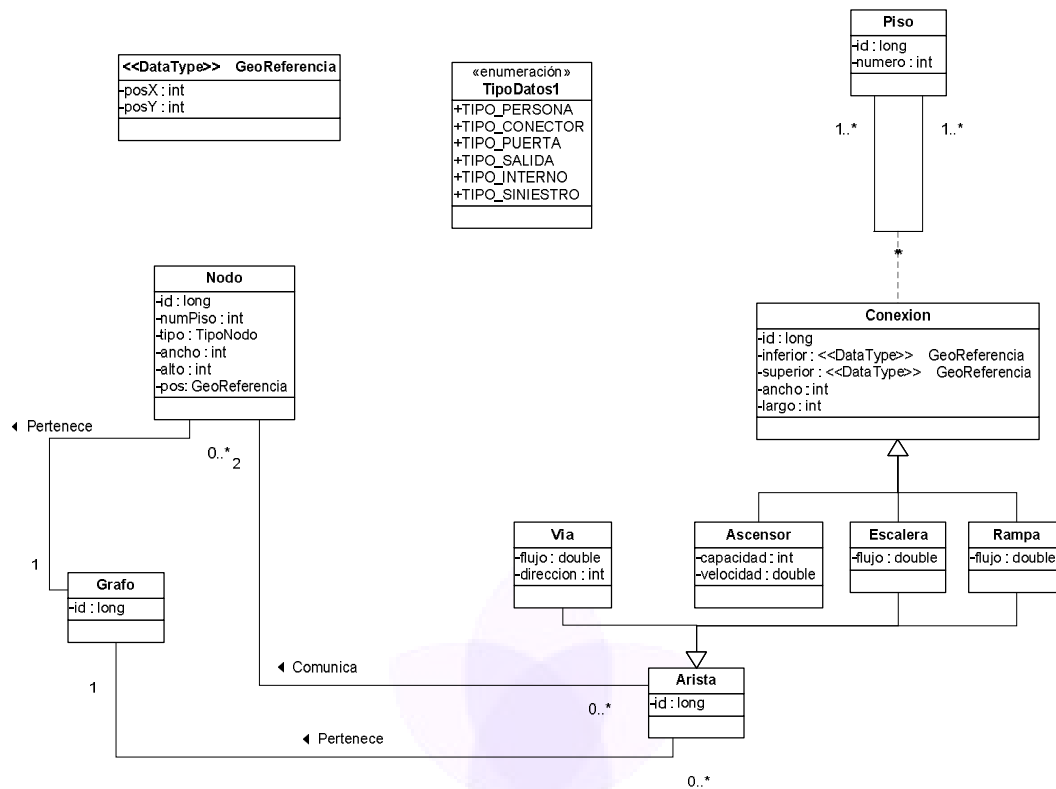


Figura 3 Diagrama de clases de Grafo

Definida la representación de la matriz y del grafo, se explicará a continuación la manera en que se genera un grafo a partir de la matriz de celdas.

Notación de tipos de celdas

En los siguientes ítems se utilizan figuras de matrices para ejemplificar el algoritmo. En las mismas se adoptan diferentes colores para identificar los tipos de celdas:

- Personas: azul.
- Salida: verde.
- Puerta interna: verde claro.
- Pared: marrón.
- Libre o vacía: celeste.
- Nodo interno: amarillo
- Arista: gris

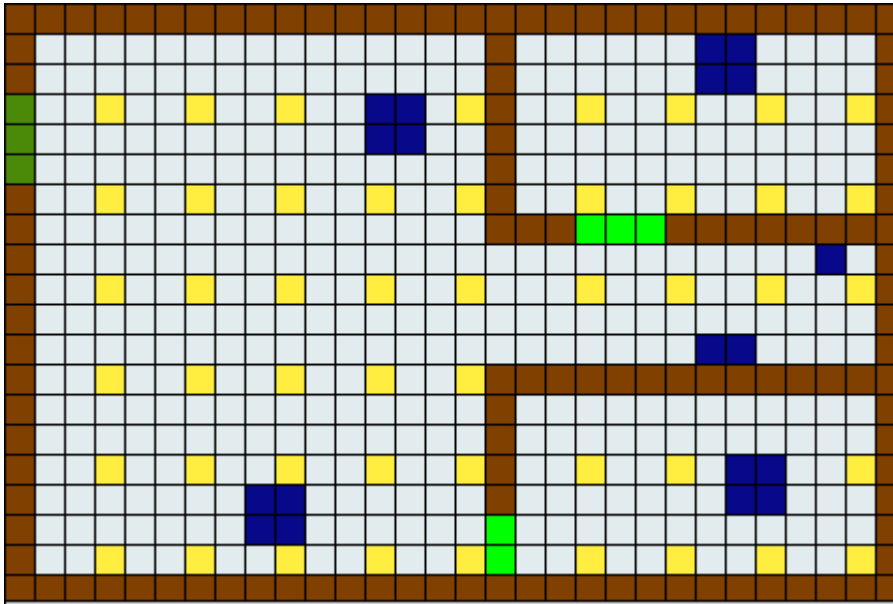


Figura 6 Distribución de los nodos internos

Aristas

Las aristas de tipo Escalera, Rampa y Ascensor se generan directamente a partir de las clases que las representan.

Por otro lado, las aristas de tipo Vía surgirán a partir de las búsquedas de caminos directos entre los nodos ya definidos anteriormente (internos, conexiones, puertas internas, salidas y personas). Existirá una arista de tipo Vía entre dos nodos cuando exista un camino directo entre ellos, o sea, cuando no existan obstáculos que impidan su unión en alguna de las posibles direcciones (horizontales, diagonales o verticales), las cuales se detallarán más adelante continuación.

Cabe agregar que las aristas podrán ser transitadas en cualquier sentido. Para contemplar esto, entre dos nodos que estén unidos se crean dos aristas con sentido contrario.

Si en la búsqueda de nodos adyacentes se encuentra una arista ya existente, entonces se divide dicha arista, creando un nuevo nodo intermedio en el punto de intersección. Este nuevo nodo será el adyacente.

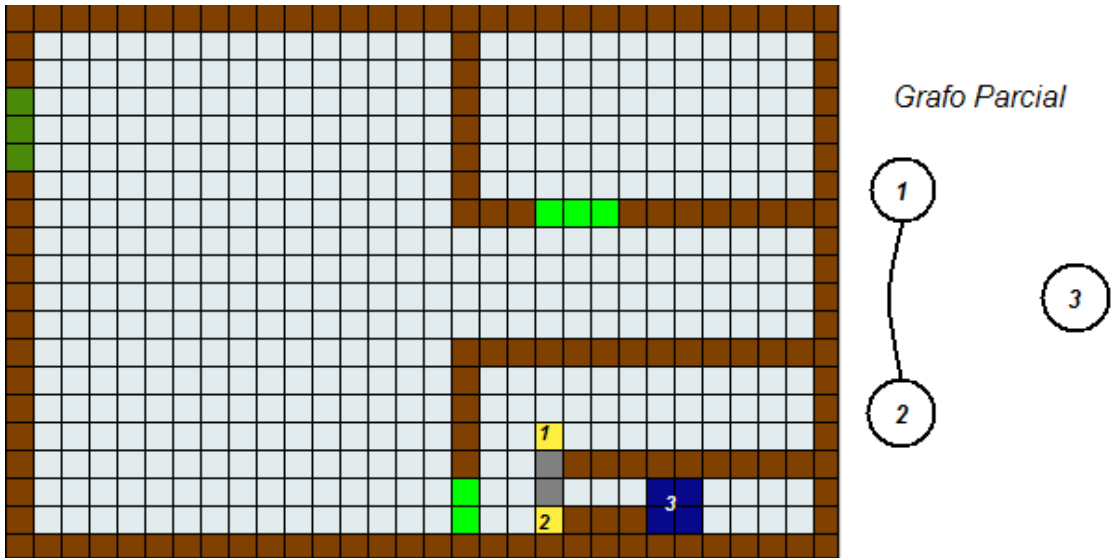


Figura 7 Matriz y grafo parcial antes de encontrar y dividir una arista

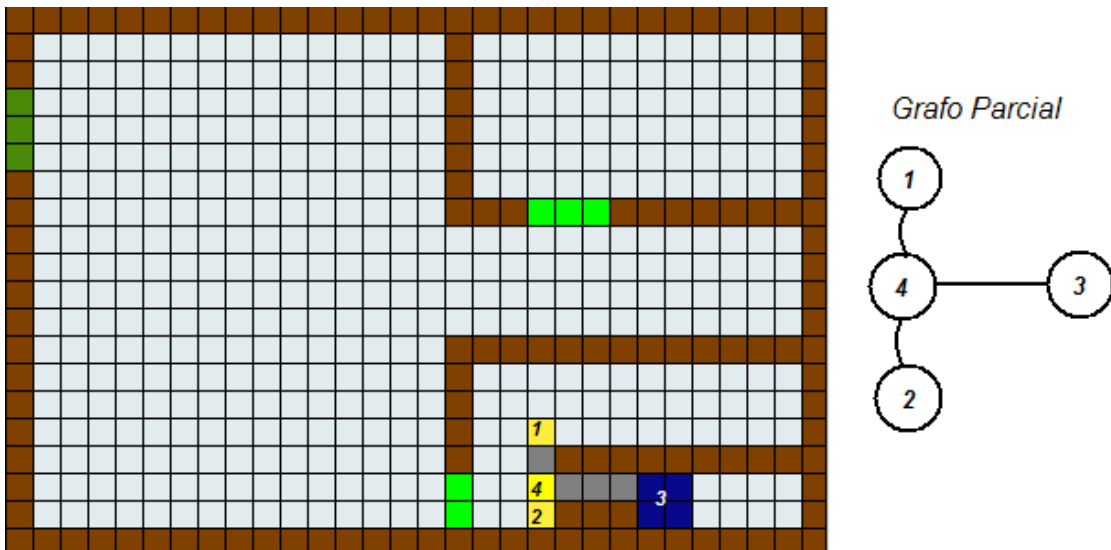


Figura 8 Matriz y grafo parcial después de encontrar y dividir una arista

Direcciones de Movimiento

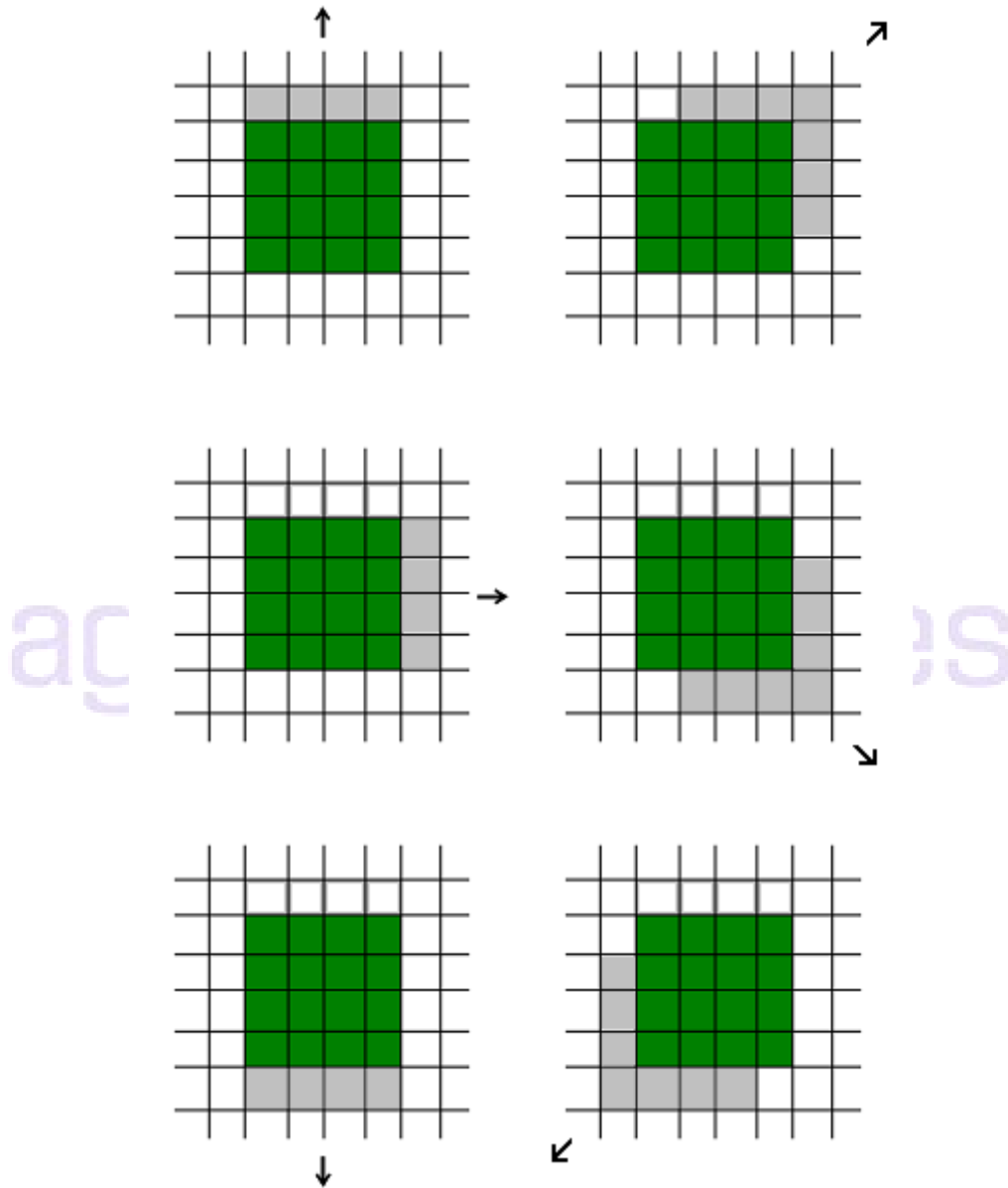
En la generación de las aristas del grafo, se intenta mover cada nodo en diferentes direcciones, buscando encontrar caminos que lo puedan llevar hacia otros nodos. Si se encuentra un camino, entonces se agrega la arista correspondiente en el grafo.

Es así que se definen 8 direcciones de movimiento:

- 1: hacia arriba
- 2: hacia arriba-derecha
- 3: hacia derecha
- 4: hacia abajo-derecha

- 5: hacia abajo
- 6: hacia abajo-izquierda
- 7: hacia izquierda
- 8: hacia arriba-izquierda

En la figura 9 se muestra gráficamente como es el movimiento de un nodo en cada dirección.



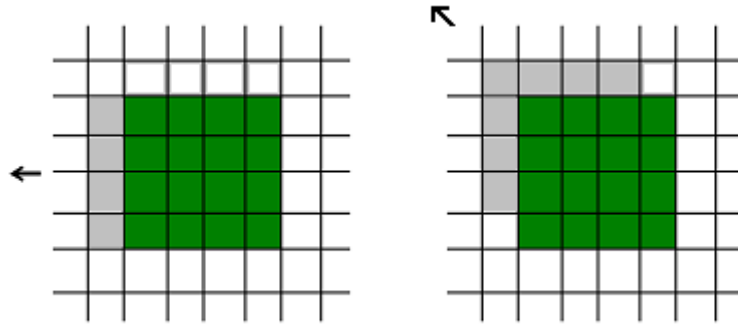


Figura 9 Posibles direcciones de movimiento de un nodo

Movimientos de nodos

Dado que los nodos están formados por un rectángulo de celdas contiguas, muchas veces el camino no estará totalmente libre. Es decir, puede suceder que las siguientes celdas del avance en una dirección no estén todas libres, sino que existan obstáculos (pared, zona intransitable). Si los obstáculos comprenden todas las siguientes celdas, entonces no se podrá avanzar, pero si son sólo algunas de las siguientes celdas, el avance continuará (aunque disminuirá el flujo de movimiento).

Para contemplar estos casos es que, en la búsqueda de caminos, se utilizan "nodos dinámicos", los cuales podrán achicarse (disminuyendo el alto y/o ancho) o agrandarse (aumentando el alto y/o ancho) según sea necesario. Es así que los "nodos dinámicos" tendrán alto y ancho inicial, y alto y ancho actual. Los valores actuales del alto y ancho en ningún momento podrán superar los iniciales.

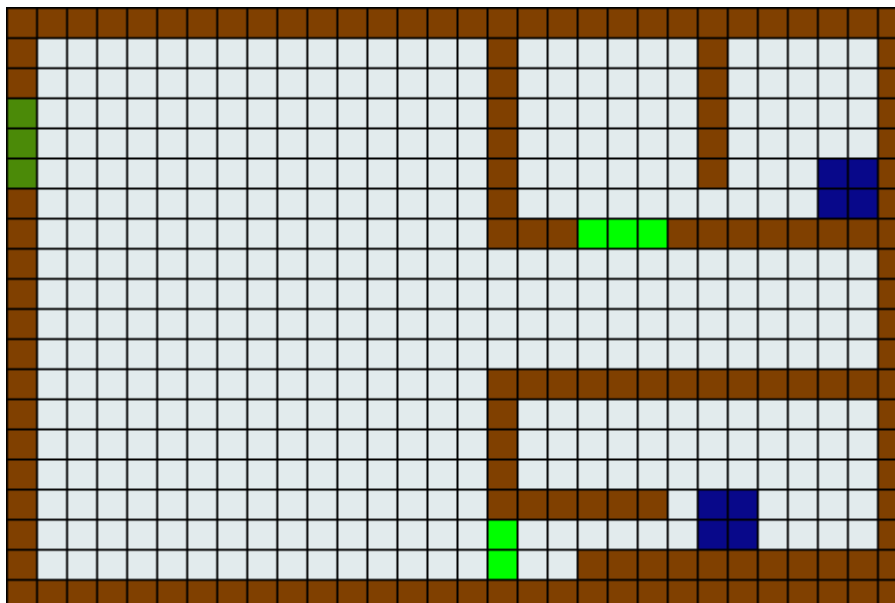


Figura 10 Ejemplo en el que es necesario "achicar" y luego "agrandar" los nodos

Pseudocódigo

Para comprender mejor el algoritmo de determinación de las aristas de tipo Vía, se detallará el pseudocódigo simplificado de dos funciones empleadas para definir las aristas de cada nodo.

La función auxiliar *buscarNodoAdyacente* busca un nodo adyacente en una dirección determinada, devolviendo el identificador del nodo encontrado. Como se dijo anteriormente, en caso de encontrar una arista ya existente, se divide dicha arista, creando un nuevo nodo intermedio en el punto de intersección. El identificador de este nuevo nodo será el retornado.

Si no existen nodos adyacentes en la dirección dada, la función retorna -1.

```

buscarNodosAdyacente(matriz: Matriz, grafo: Grafo, nodo: Nodo, dir: int): int
    int idNodoAdy = -1
    Nodo nodoDinamico = clonar(nodo)
    boolean encuentreNodo = false
    boolean encuentreArista = false
    Mientras (not encuentreNodo AND puedeAvanzar(matriz,nodo,dir)) entonces
        avanzarNodo(matriz,nodo,dir)
        encuentreNodo = llegaANuevoNodo(matriz,nodoDinamico)
        Si (encontreNodo) entonces
            idNodoAdy = getIdNodoAdy(matriz,nodoDinamico)
        En otro caso
            encuentreArista = llegaAArista(matriz,nodoDinamico)
            Si (encontreArista) entonces
                // Se divide la arista y devuelve el identificador
                // del nuevo nodo "intermedio" creado
                idNodoAdy = dividirArista(matriz,grafo,nodoDinamico)
        Fin Si
    Fin Si
    return idNodoAdy
Fin

```

La función *agregarAristas*, utiliza la función *buscarNodoAdyacente* para buscar los nodos adyacentes en todas las direcciones y agregar las aristas correspondientes en el grafo.

Observar que en ambas funciones se utilizan métodos iterativos (*Para cada*, *Mientras*).


```

agregarAristas(matriz: Matriz, grafo: Grafo, nodo: Nodo)
    Nodo nodoAdy;
    Para cada (dir = 1, ..., 8)
        idNodoAdy = buscarAdyacente(matriz, grafo, nodo, dir)
        Si (idNodoAdy >= 0) entonces
            nodoAdy = getNodo(grafo, idNodoAdy)
            agregarArista(matriz, grafo, nodo, nodoAdy)
        Fin Si
    Fin Para
Fin

```

4.2.2 Rutas óptimas

Para el cálculo de rutas óptimas de evacuación y hacia las zonas siniestradas, se empleo el algoritmo de Dijkstra[5] con algunas modificaciones.

El algoritmo de Dijkstra, también llamado algoritmo de caminos mínimos, es un algoritmo para la determinación del camino de costo mínimo dado un vértice origen al resto de vértices en un grafo dirigido y con pesos en cada arista. El algoritmo tiene como precondition que no existan ciclos de costo negativo, para evitar que los caminos óptimos puedan pasar dos veces por el mismo vértice. En este caso el algoritmo es aplicable, ya que el peso de cada arista está dado por la distancia entre los nodos extremos, que siempre es mayor o igual que cero, por lo que no existen ciclos negativos. Por motivos de simplicidad, la unidad de distancia utilizada es la celda. Por ejemplo, si dos nodos están separados dos celdas horizontal o verticalmente, entonces dichos nodos tendrán distancia igual a dos. Cabe agregar que el valor de distancia es un decimal, para contemplar el hecho de que existan aristas diagonales. Para el caso de las conexiones (escaleras, rampas, ascensores), el usuario ingresa el largo de las mismas en centímetros, por lo que es necesario realizar la transformación de distancia en centímetros para la distancia en celdas, según la escala (correspondencia entre celdas y centímetros) utilizada.

La idea subyacente en este algoritmo consiste en ir explorando todos los caminos más cortos que parten del vértice origen y que llevan a todos los demás vértices; cuando se obtiene el camino de costo mínimo desde el vértice origen, al resto de vértices que componen el grafo, el algoritmo se detiene.

También se maneja el concepto de nodos habilitados o deshabilitados. El algoritmo sólo recorre los nodos habilitados. En el caso del cálculo de las rutas de evacuación, los nodos de tipo siniestros estarán deshabilitados para evitar el

pasaje por esos lugares. En el caso de la búsqueda de caminos hacia él/los siniestro/s, los mismos estarán habilitados.

Pseudocódigo del Dijkstra modificado

```

Procedimiento Dijkstra()
  Para k=0 a cantNodos
    Si es nodohabilitado(k)
      Si existeArista(nodoInicial,k)
        Dist[k]= distArista(nodoInicial,k)
        Camino[k]=nodoInicial
      Sino
        Dist[k]=infinito
        Camino[k]=-1
  Mientras haya que seguir...
    Para k=0 a cantNodos
      Si Dist[k]<= minimo & (nofuevisitado[k])
        Proximo=k
        Minimo=Dist[k]
    Para k=0 a cantAristas
      Si es nodohabilitado(destino)
        Si (Dist[destino]>Dist[proximo] + distArista(proximo,destino))
          Dist[destino]=Dist[proximo]+distArista(proximo,destino)
          Camino[destino]= proximo
    NoSeguir
  Para k=0 a cantNodos
    Seguir=seguir || nofuevisitado[k]

```

Para encontrar las rutas óptimas de evacuación, los nodos origen son los grupos de personas, por lo que se aplica el algoritmo dijkstra para cada uno de ellos. De esta manera se encuentran los caminos óptimos hasta los nodos de tipo salida.

Para el caso de las rutas óptimas hacia los siniestros se consideró el hecho que el grafo no es dirigido (o en otras palabras, siempre que exista una arista desde un nodo a hasta b , entonces existirá una arista desde b hasta a con el mismo peso), entonces la ruta óptima desde un nodo n_1 hasta n_2 es igual que la ruta óptima desde el nodo n_2 hasta n_1 pero en sentido contrario. De esta manera, como en general la cantidad de siniestros es mucho menor que la cantidad de salidas, se decidió utilizar los siniestros como nodos origen, disminuyendo la cantidad de veces que se ejecuta el algoritmo. Los caminos óptimos serán los encontrados pero en sentido contrario, o sea desde cada salida hasta cada siniestros.

4.2.3 Zonas de Vulnerabilidad

Esta sección se divide en tres partes. En las dos primeras, se explican brevemente los conceptos de *Ptos. de Articulación* y *Búsqueda en Profundidad* sobre grafos. Estos conceptos son manejados para la *Determinación de los niveles de vulnerabilidad*, que se detalla en la tercera parte.

4.2.3.1 Puntos de Articulación

Un grafo dirigido es llamado fuertemente conexo si para cada par de vértices u y v existe un camino de u hacia v y un camino de v hacia u . Los componentes fuertemente conexos (CFC) de un grafo dirigido son sus subgrafos máximos fuertemente conexos. Estos subgrafos forman una partición del grafo.

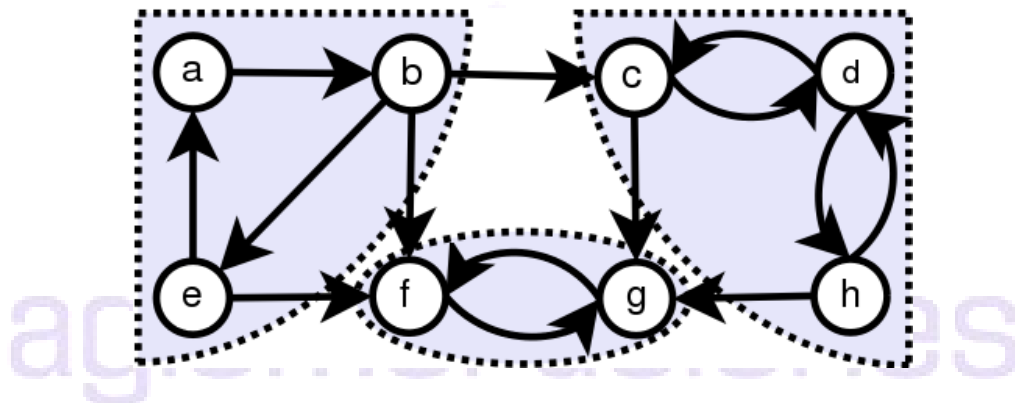


Figura 11 Componentes fuertemente conexos

Sea $G = (V, E)$ un grafo dirigido:

1. Aplicar búsqueda en profundidad sobre G
2. Calcular el grafo traspuesto.
3. Aplicar búsqueda en profundidad sobre G^t (el grafo traspuesto) iniciando la búsqueda en los nodos de mayor a menor tiempo de finalización obtenidos en la primera ejecución de búsqueda en profundidad (paso 1)
4. El resultado será un bosque de árboles. Cada árbol es una componente fuertemente conexa.

Los componentes fuertemente conexos del grafo, tienen cabida en la determinación de las zonas vulnerables.

4.2.3.2 Búsqueda en profundidad

Algoritmo que permite recorrer todos los nodos de un grafo o árbol de manera ordenada, pero no uniforme. Su funcionamiento consiste en ir expandiendo todos y cada uno de los nodos que va localizando, de forma recurrente, en un camino concreto. Cuando ya no quedan más nodos que visitar en dicho camino, regresa (Backtracking), de modo que repite el mismo proceso con cada uno de los hermanos del nodo ya procesado.

```
DFS(grafo G)
  PARA CADA vertice  $u \in V[G]$  HACER
    estado[u] ← NO_VISITADO
    padre[u] ← NULO
    tiempo ← 0
  PARA CADA vertice  $u \in V[G]$  HACER
    SI estado[u] = NO_VISITADO ENTONCES
      DFS_Visitar(u)
```

```
DFS-Visitar(nodo u)
  estado[u] ← VISITADO
  tiempo ← tiempo + 1
  d[u] ← tiempo
  PARA CADA  $v \in Vecinos[u]$  HACER
    SI estado[v] = NO_VISITADO ENTONCES
      padre[v] ← u
      DFS_Visitar(v)
  estado[u] ← TERMINADO
  tiempo ← tiempo + 1
  f[u] ← tiempo
```

4.2.3.3 Determinación de los niveles de vulnerabilidad

Los niveles de vulnerabilidad se determinan asignando valores de riesgo a los nodos y aristas del grafo. Cuanto mayor sea el riesgo de un nodo o aristas, mayor será la vulnerabilidad de la zona comprendida por los mismos.

En la determinación de los niveles de riesgos se consideran dos factores. Por un lado, la cercanía a las zonas siniestradas, y por otro lado el acceso a las salidas.

Se realizan tres pasos:

- 1) Determinación de riesgo de nodos según la cercanía a siniestros
- 2) Determinación de riesgo de nodos según el acceso a salidas
- 3) Determinación de riesgo de aristas

1) Determinación de riesgo de nodos según la cercanía a siniestros

Primeramente se definen los niveles de los nodos según la distancia a los siniestros, determinando los siguientes valores:

NIVEL DE RIESGO	DISTANCIA A SINIESTRO (cm)
3	0 A DIST_ENTRE_NIVELES
2	DIST_ENTRE_NIVELES A (2 * DIST_ENTRE_NIVELES)
1	(2 * DIST_ENTRE_NIVELES) A (3 * DIST_ENTRE_NIVELES)

Aquí, la constante DIST_ENTRE_NIVELES se define de acuerdo a la intensidad del siniestro ingresada por el usuario, teniendo la siguiente correspondencia:

INTENSIDAD DEL SINIESTRO	DIST_ENTRE_NIVELES (cm)
Baja	300
Media	500
Alta	700

2) Determinación de riesgo de nodos según el acceso a salidas

Luego se determinan los niveles de los nodos según el acceso a salidas. Aquí los valores se definen según los pto. de articulación (puertas, conexiones) que se visitan para llegar a la salida más cercana.

NIVEL DE RIESGO	CANTIDAD DE PTOS. DE ARTICULACIÓN VISITADOS
0	0 (Salida Directa)
1	1
2	2
3	3 o más
MAX_RIESGO	SIN SALIDA

Notar que, para los nodos que no tienen acceso a salidas se asigna un valor de riesgo máximo (MAX_RIESGO).

En el caso de que a un nodo ya se le haya asignado un riesgo en el paso anterior, entonces se sobrescribirá únicamente si el nuevo valor calculado en este paso es mayor.

Otro punto a tener en cuenta es que, si en los caminos hacia las salidas existe un nodo con nivel de riesgo mayor, entonces se tomará este último valor para el nodo actual. Por ejemplo, si un nodo tiene salida directa (riesgo 0), pero debe atravesar por un nodo con riesgo mayor (debido a la cercanía a un siniestro) para llegar a la salida, entonces se le asignará este último valor.

3) Determinación de riesgo de aristas

Después de haber definido los niveles de riesgo de los nodos, se asignan los niveles a las aristas, tomando el mayor riesgo de los nodos extremos de cada arista.

$\text{RIESGO (Arista)} = \text{MAXIMO} \{ \text{RIESGO}(\text{NodoOrigen}), \text{RIESGO}(\text{NodoDestino}) \}$

De esta manera se realiza una "trasposición" de los niveles de riesgo calculados primeramente en base a la cercanía a siniestros y luego en base al acceso a las salidas.

En la figura 12 se muestra un ejemplo de los niveles de vulnerabilidad generados de acuerdo a lo detallado anteriormente.

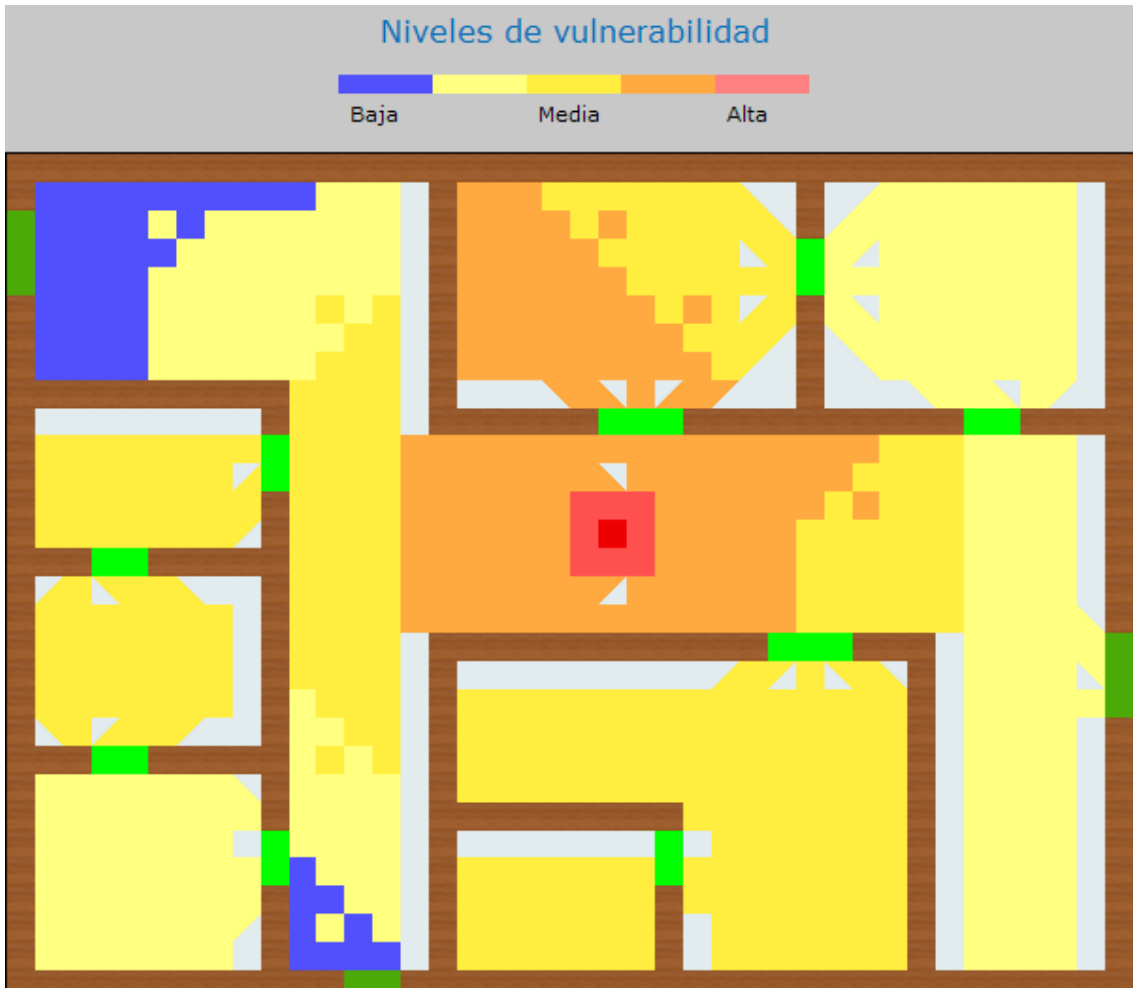


Figura 12 Niveles de Vulnerabilidad

4.2.4 Puntos Críticos

Se dice que un vértice de un grafo conexo es un punto de articulación del mismo si al suprimir ese vértice y todas las aristas que inciden en él, el grafo resultante deja de ser conexo. Así mismo, un puente es un arco de G cuya eliminación deja al grafo resultante disconexo[6].

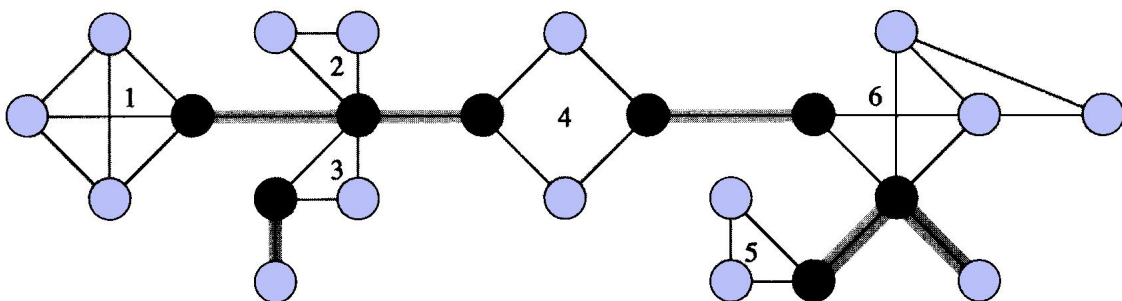


Figura 13 Puntos de Articulación

A cada nodo se le asigna un número de visita v , y es identificado por él.

$Low[v] = \min\{v \cup w \mid \exists(x,w) \in B\}$, donde:

x es descendiente de v , y w es ancestro de v

v es un *punto de articulación* si no es raíz y tiene un hijo s tal que $Low[s] \geq v$

Para calcular $Low[v]$ se requiere revisar toda la lista de adyacencia de v .

$Low[v] = \{ \min\{v \cup w \mid \exists (v,w) \in B \cup Low[s]\}$, donde: s es hijo de v

```

procedure SEARCHB( $v$ ) {
  marca  $v$  "visitado";
   $DFNUMBER[v] \leftarrow COUNT$ ;
   $COUNT \leftarrow COUNT + 1$ ;
   $LOW[v] \leftarrow DFNUMBER[v]$ ;
  for cada vértice  $w$  en  $L[v]$  do
    if  $w$  está marcado "no visitado" {
      agrega  $(v,w)$  to  $T$ ;
       $FATHER[w] \leftarrow v$ ;
      SEARCHB( $w$ );
      if  $LOW[w] \geq DFNUMBER[v]$ 
        un componente biconectado se encontró;
       $LOW[v] \leftarrow \min(LOW[v], LOW[w])$ 
    }
    else if  $w$  no es  $FATHER[v]$  then
       $LOW[v] \leftarrow \min(LOW[v], DFNUMBER[w])$ 
}

```

Los puntos de articulación (o vértices puentes) del grafo, representan los puntos críticos del plano, es decir aquellos puntos o zonas transitables de máxima prioridad, que ante la ocurrencia de un siniestro, pueden dejar una o varias zonas del plano inalcanzables.

5 Diseño

En esta sección se muestra el diseño del prototipo implementado. Para la visualización se emplea el diagrama de alto nivel de la arquitectura (ver Documento de Arquitectura, en capítulo Anexos) y el diagrama de clases (ver Documento de Casos de Uso, en capítulo Anexos).

5.1 Consideraciones generales de diseño

El diseño de un sistema es consecuencia directa de los requerimientos y la infraestructura, tanto de software como de hardware, que se dispone.

Es primordial comprender claramente los requerimientos, para luego buscar alternativas de diseño que contemplen y se adapten a ellos. En este análisis es conveniente comenzar por los requerimientos no funcionales, ya que son características generales del sistema, que se deben aplicar en todas las funcionalidades.

Es así, que algunas características importantes a considerar son:

- **Usabilidad:** el sistema debe ser fácil de usar, eficiente tanto en el tiempo como en la calidad de las respuestas.
- **Generalización:** a pesar de que se aplicará a un caso y lugar concreto, el sistema debe ser diseñado de tal manera que pueda ser utilizado en otros tipos de locales similares.
- **Mantenimiento y adaptabilidad:** la aplicación debe ser flexible para su adaptación a nuevas situaciones futuras, y poder ser mantenida por personas previamente capacitadas. O sea que, la idea es realizar un producto que esté en constante mejora e incorporación de funcionalidades.
- **Riesgo cero:** los algoritmos y soluciones deben poseer el concepto de "riesgo cero". Es decir que las respuestas críticas del sistema no deberían fallar. Esto es un requerimiento difícil de implantar, para el cual es fundamental un buen diseño y testeado.

5.2 Arquitectura

Dado que el sistema será utilizado por variados y diferentes usuarios, es conveniente pensar en una aplicación centralizada pero que pueda ser accedida desde varias máquinas. En otras palabras, para que el sistema sea usable, varios usuarios deben poder trabajar y acceder a la misma información, pero

desde distintas máquinas. Es por ello, que se optó en definir una aplicación web, aprovechando las ventajas y el constante crecimiento que están teniendo estas tecnologías.

De esta forma, se define un servidor, en el que se encuentra la aplicación, y al cual accederán varios usuarios, lo que es una típica aplicación cliente/servidor.

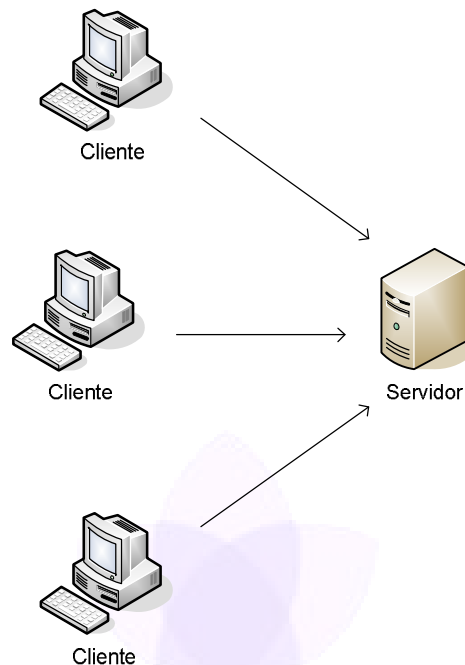


Figura 14 Aplicación Cliente/Servidor

Corresponde ahora, definir la arquitectura del servidor de aplicación. A modo de independizar la interfaz de presentación, el procesamiento de las solicitudes y la persistencia de los datos, se define una arquitectura en tres capas bien diferenciadas: *Presentación*, *Lógica* y *Base de Datos*. Esta división ayuda a que el sistema pueda ser fácilmente mantenido y adaptado a futuras mejoras, ya que, por ejemplo, sería posible cambiar la interfaz de presentación sin afectar la lógica, o viceversa.

Capa de Presentación

Es la interfaz de la aplicación, o sea la manera en que el usuario percibe e interactúa con el sistema. Es independiente de la lógica, es decir que, se puede cambiar la presentación sin afectar la manera en que se procesan de solicitudes y los resultados retornados. Si bien la estética no es el objetivo del proyecto, es bueno que en un futuro pueda ser mejorado este aspecto, sin afectar la lógica.

Esta capa se comunica con la capa lógica, enviando solicitudes y recibiendo los resultados correspondientes.

Capa Lógica

Es la capa fundamental de la aplicación, donde se procesan los datos y algoritmos. Aquí es donde toman mayor relevancia los conceptos de generalización, mantenimiento, adaptabilidad y "riesgo cero". Es la capa intermedia, por lo que interactúa con la presentación para recibir las solicitudes y retornar los resultados, y con la base de datos para persistir y consultar datos.

Capa de Persistencia

Se corresponde con la base de datos, o sea, donde serán persistidos los datos una vez confirmados por el usuario, y luego consultados. Es la capa de más bajo nivel, e interactúa con la capa lógica para retornar resultados de una consulta o persistir datos recibidos.

Para la comunicación entre las capas, dentro de cada una de ellas se definen interfaces de acceso. A continuación se muestra un esquema de la arquitectura descrita.

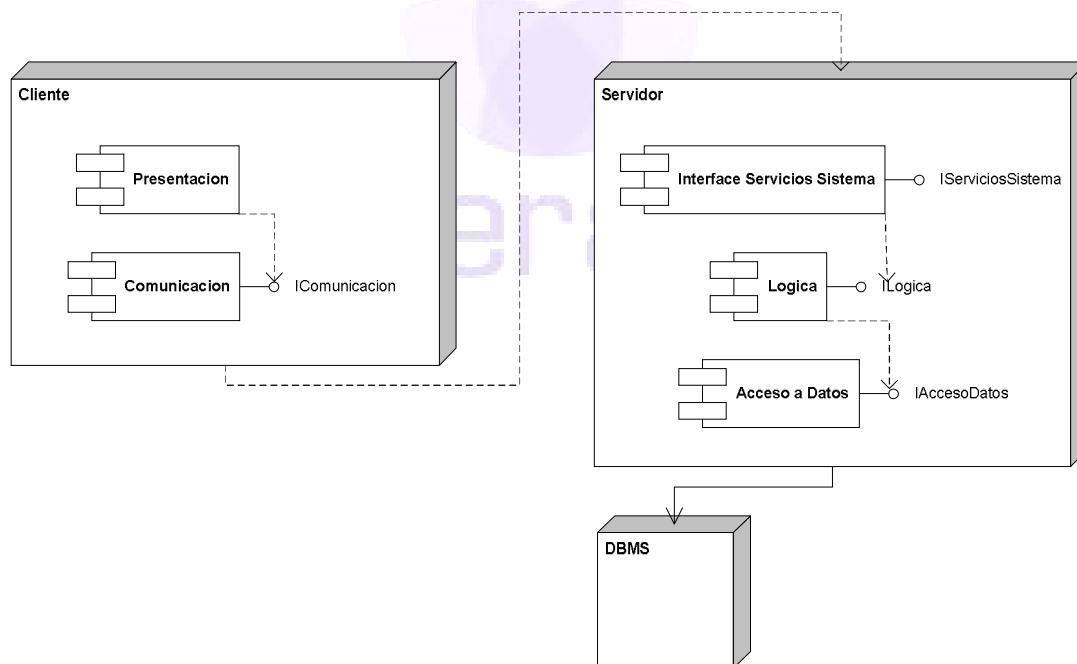


Figura 15 Arquitectura (Alto Nivel)

5.3 Diagrama de Clases

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.

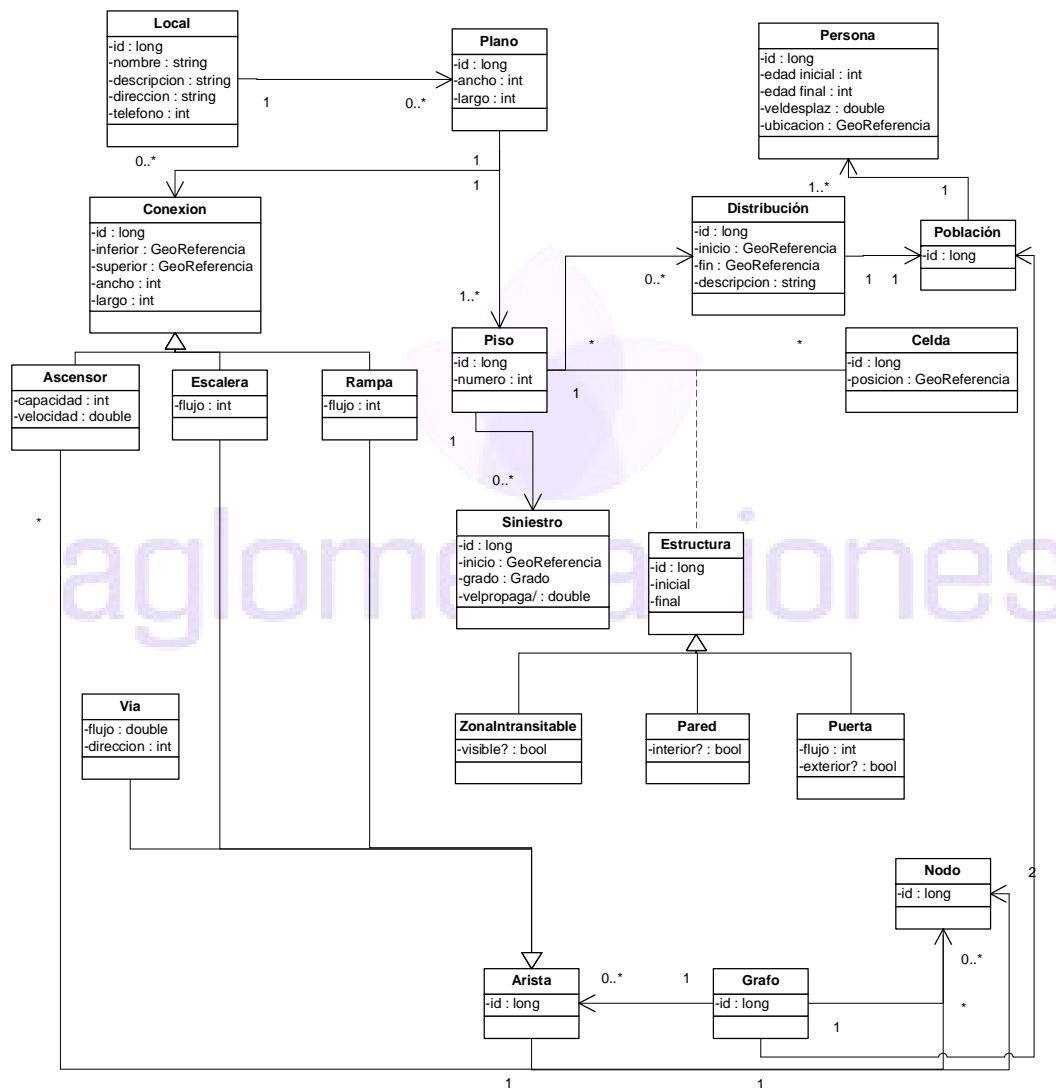


Figura 16 Diagrama de Clases

Describimos a continuación cada una de las entidades junto con sus atributos.

Local: representa el local (shopping, terminal, centro comercial) modelado en el sistema, conteniendo información básica del mismo: nombre, descripción, dirección y teléfono.

Plano: representa el plano completo asociado al local modelado. Contiene un conjunto de pisos y conexiones que unen los pisos. Los atributos de esta entidad son el largo y ancho, que definen el área de los pisos que componen el plano. En el modelo, todos los pisos tienen igual área. Si existe un piso de menor área, se definirán zonas intransitables en los sectores laterales.

Piso: representa un determinado piso del plano, especificado por el atributo *número*. El piso con número cero indica la planta baja, y los números negativos, los subsuelos. Se debe controlar que los pisos sean consecutivos, es decir, que si existe el número 3 debe existir el 2, 1 y 0.

Conexión: representa la conexión entre dos pisos, y puede ser de tipo *Escalera*, *Rampa* o *Ascensor*. En caso que unan más de dos pisos (*Ascensor*), existirá una entidad por cada unión. Los atributos comunes son los puntos de conexión entre el piso superior e inferior, el largo y ancho de la conexión. En el caso de la *Escalera* y la *Rampa* se especifica el flujo, mientras que para el *Ascensor* se especifica velocidad y capacidad.

Estructura: representa un elemento o componente del plano de un piso. Puede ser una *Zona Intransitable*, *Pared* o *Puerta*. Los atributos comunes son la posición inicial y final, que determinan el área rectangular en el cual está definido el elemento.

Distribución: representa una distribución determinada de personas (por ejemplo "Viernes de noche", "Feriado"). Cabe decir que, cuando el usuario crea una distribución, se crea una entidad *Distribución* vacía para cada piso. Posteriormente, el usuario irá definiendo la ubicación de las personas de la misma en cada piso, y se van agregando en el modelo.

Población: representa la población de personas en un determinado piso para una distribución definida.

Persona: representa un conjunto de personas que se estima que estén presentes en un sector del local en determinado momento (*Distribución*). Para las mismas se especifica la ubicación y velocidad.

Siniestro: representa un siniestro en el local. Esta identidad se crea cuando se indica su ocurrencia, especificando la ubicación inicial, el grado (importancia, gravedad), y la velocidad estimada de propagación (en caso de incendio por ejemplo).

Grafo: representa el grafo a utilizar en el momento de la simulación, para determinar los caminos óptimos. Esta entidad se crea cada vez que se realiza la simulación, a partir del plano del local y la distribución de personas.

Arista: representa una arista del grafo.

Nodo: representa un nodo del grafo.

Vía: representa un sector del plano por el que podrán circular las personas para dirigirse hacia las salidas.

En el siguiente diagrama de clases se muestra gráficamente las entidades y sus relaciones.



6 Implementación

En esta sección se detalla el diagrama de deployment utilizado en el prototipo implementado, así como las tecnologías utilizadas para la construcción del mismo.

6.1 Diagrama de Implementación

Los Diagramas de Implementación se usan para modelar la configuración de los elementos de procesamiento en tiempo de ejecución y de los componentes, procesos y objetos de software que viven en ellos. Así, este diagrama es un buen lugar para mostrar de manera macro, el recorrido de los datos, dentro de un sistema distribuido.

Cada **nodo** del diagrama representa una unidad de cómputo; se puede tratar de hardware distinto o podría ser el mismo.

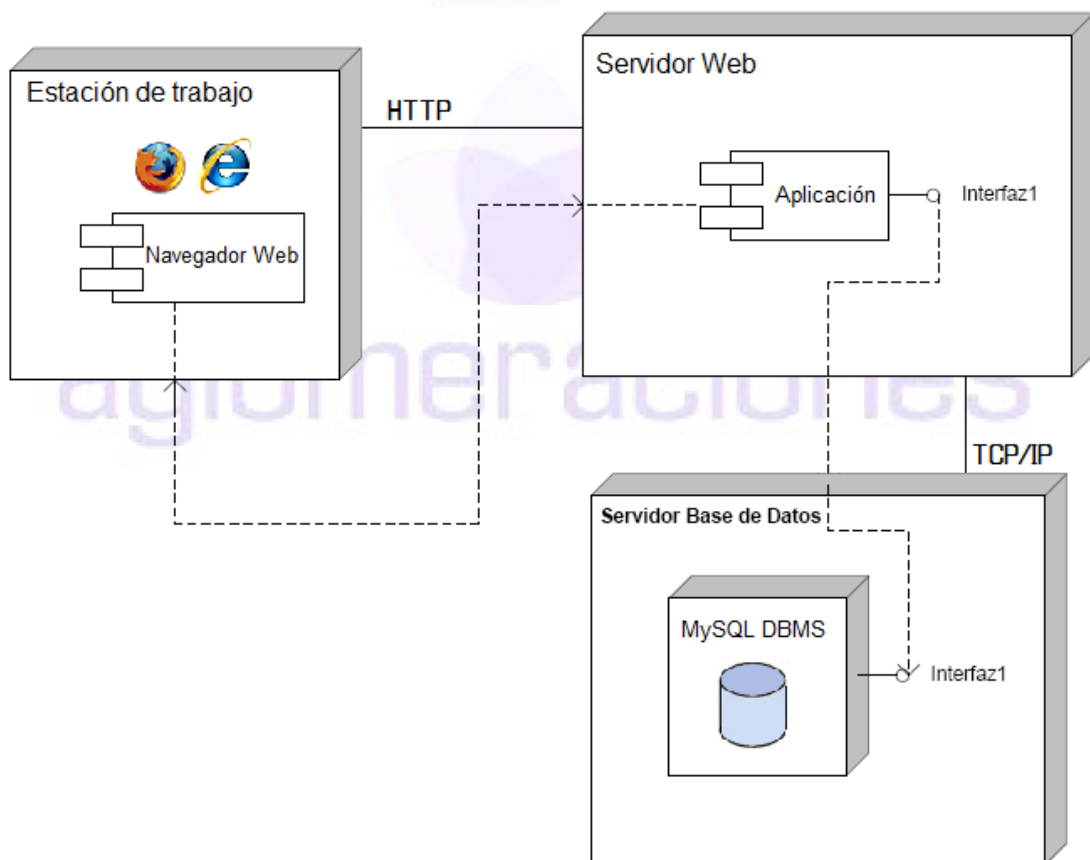


Figura 17 Diagrama de deployment

En la figura 17 vemos como el usuario desde su estación de trabajo, accede a la aplicación, usando su navegador web. Esta conexión se logra vía HTTP. Las conexiones entre nodos muestran los mecanismos de comunicación a través de las cuales interactuará el sistema. La aplicación, que está alojada en el servidor Web, se comunica con la Base de Datos (MySQL) mediante el protocolo TCP/IP. Durante el desarrollo del sistema usamos un servidor ágil como lo es Jetty, pero para la instalación final elegimos Apache Tomcat por su robustez.

Los componentes del diagrama representan módulos físicos de código. Estos se corresponden con a los paquetes de un diagrama de paquetes que se incluye en el diseño, de esta manera el diagrama nos muestra dónde se ejecuta cada paquete en el sistema. En este caso nuestro código sería el que está en el servidor Web, ya que el Navegador es tan solo un cliente fino y la BD nos provee la persistencia de datos, pero no tenemos programados triggers ni stored procedures.

Las dependencias entre los componentes muestran cómo se comunican los componentes entre sí. La dirección de una dependencia dada indica el conocimiento en la comunicación.

6.2 Tecnologías Empleadas

6.2.1 Framework de Desarrollo

Para el desarrollo de proyecto, se buscó un framework que se adaptara a la arquitectura definida, también, que sea eficiente, fácil de utilizar y aproveche las últimas innovaciones de desarrollo web.

Basados en nuestra experiencia previa, optamos por la utilización de *Grails*[10] para el desarrollo de la presentación, lógica e interacción con la base de datos, dado que cumple con características deseadas, descritas en el párrafo anterior.

Grails es un framework para aplicaciones web (acorde a la arquitectura), open source (ideal para una entidad gubernamental), que aprovecha el lenguaje de programación *Groovy*[11] (tecnología de avanzada, que a su vez está basado en la plataforma Java).

Entre sus características y ventajas se destacan:

- *Alta productividad*
- *Extensible*
- *Reuso de tecnologías ya consagradas en Java como Hibernate y Spring*
- *Facilidad de aprendizaje*
- *Buen ambiente de desarrollo*
- *Open source*
- *Documentación accesible y fácilmente comprensible*

Además, dado que está basado en Java, es posible interactuar entre *Grails* y clases Java.

Se decidió entonces, que los algoritmos de generación de grafo y búsqueda de rutas óptimas fueran realizados en clases Java independientes, que se llaman desde el proyecto web en el momento de la simulación. Se logró así una independencia interesante, ya que se pudo desarrollar y modificar los algoritmos en las clases Java correspondientes, sin necesidad de realizar cambios en el resto del código.

6.2.2 Servidor de aplicaciones

Durante la fase de implementación se empleó el Servidor de aplicaciones Derby[12] que viene embebido con el paquete GRAILS, cuya principal ventaja para la implementación es el HotDeploy (es la habilidad de compilar y cargar en memoria los archivos compilados en tiempo de ejecución). Esta funcionalidad permite testear durante el desarrollo, sin necesidad de regenerar cada vez que se produce un cambio en el código fuente.

Esto facilitó la tarea de desarrollo y pruebas unitarias.

Sin embargo, para la implantación del prototipo, se optó por la utilización de Tomcat[13] que si bien no es un servidor de aplicaciones, es un servidor web con soporte de servlets y JSPs (utilizados en el desarrollo del proyecto).

6.2.3 Manejador de base de datos

Como manejador de base de datos se optó por MySQL[14], dado que el mismo es Open Source, de fácil manejo e instalación, además de ser ampliamente reconocido.

6.2.4 WebServices

El sistema Aglomeraciones en el módulo de diseño, al crear el Local, se integró con Google Maps[15] por medio de webservices para obtener la ubicación del mismo. Este dato será indispensable para que en trabajos futuros se consideren el acceso de los vehículos de emergencia y traslado de evacuados hacia los centros asistenciales más adecuados.

6.2.5 Ajax

En las simulaciones de siniestro se emplearon técnicas AJAX, que están integradas en el framework *Grails*. Esta tecnología contribuye a una mayor interactividad, velocidad y usabilidad de la aplicación, ya que permite procesar datos mientras el usuario está realizando acciones.

7 Testeo

Este capítulo tiene como finalidad describir el tipo de pruebas realizadas, y presentar los resultados obtenidos a partir de las mismas.

7.1 Descripción de los test realizados

7.1.1 Test de desarrollo

Las pruebas de desarrollo son aquellas pruebas que el programador va haciendo mientras desarrolla para verificar que lo que va programando funciona. Normalmente estas son pruebas similares a las de caja blanca que se hacen de forma intuitiva con resultados aceptables.

7.1.2 Test de módulo (o test de unidad)

Los tests de módulo o tests de unidad son las pruebas de subprogramas, subrutinas o funciones que más adelante se unirán a otras para construir un programa. De esta forma, antes de crear un programa se prueban por separado las unidades que lo formaran.

Estos test fueron realizados por los desarrolladores durante la construcción del proyecto.

7.1.3 Test funcional

Este tipo de tests se enfoca como una tarea de caja negra (se confía en las tareas de test previas) y tiene como finalidad localizar discrepancias entre el programa y las especificaciones externas.

En este caso, se desarrollo un plan de pruebas (ver Documento de Plan de Pruebas, en capítulo Anexos) o test plan, que se adjunta como anexo a este documento.

7.1.4 Test de sistema

Bajo este nombre se agrupan variedad de tests que tienen como finalidad comparar el programa o sistema desarrollado, con los objetivos inicialmente planteados para demostrar que no las satisface.

Los test de sistema realizados fueron:

- Tests de volumen: tiene la finalidad de probar que el equipo no puede trabajar con el volumen de datos especificado en los objetivos.
- Tests de usabilidad: pretende localizar problemas relacionados con el factor humano o con la usabilidad.
- Test de seguridad: tienen como objetivo encontrar vulnerabilidades o agujeros que puedan poner en peligro la seguridad del sistema.

- Test de configuración: la finalidad de estos tests es demostrar que el sistema de configuración que adapta la aplicación al entorno de usuario (sistema operativo, bases de datos, dispositivos hardware etc.) presenta deficiencias o no funciona correctamente.

7.2 Resultados Obtenidos

Las pruebas de configuración se realizaron sobre los navegadores Internet Explorer (versión 7), Firefox (versión 3.6) y Chrome (versión 4.1).

Los testeos de desarrollo y de módulo fueron realizados por el programador que implementó cada módulo, con datos de prueba generados en forma manual.

El testeo funcional fue realizado por todo el equipo de proyecto, empleando el documento Plan de Pruebas (ver Anexos), para generar los datos de prueba.

Los resultados obtenidos del testeo coinciden con los resultados esperados, habiéndose hecho una validación del prototipo con el Cliente del proyecto (CECOED) a total satisfacción del mismo.



8 Conclusiones y trabajos futuros

En esta sección se exponen los resultados alcanzados tras la finalización del proyecto, se plasman las conclusiones y se nombran posibles trabajos futuros.

8.1 Resultados alcanzados

A continuación se detallan los aspectos positivos y negativos del proyecto realizado. Por negativos se hace referencia a aquellos requerimientos, que por decisión del equipo del proyecto y en acuerdo con el cliente y tutores, fueron dejados fuera del alcance del prototipo construido.

Cuando hablamos de resultados positivos del proyecto, creemos conveniente separar en resultados obtenidos de la investigación del proyecto y por otro lado los resultados alcanzados en el prototipo implementado.

Sobre la investigación se destacan los siguientes aspectos:

- Extensa documentación generada para la facultad y para el CECOED en el área Riesgos y Aglomeraciones. Se crearon documentos de Estado del Arte sobre Riesgos en general y otro más específico sobre Aglomeraciones, que reúnen toda la información recopilada sobre estos temas. Fueron el punto de partida del proyecto y quedan como referencia para futuros trabajos que se quieran realizar en el área.
- Aplicación de los conocimientos aprendidos durante la carrera. En este proyecto cuyo desarrollo ha llevado un tiempo de más de un año, se aplicaron muchos de los conocimientos adquiridos en la carrera. Por mencionar algunos, fueron usados algoritmos de rutas óptimas aprendidos en segundo año de facultad, estos algoritmos fueron implementados en el lenguaje de programación Java, que se estudia en el tercer año y se aplicaron los conceptos de Ingeniería de Software correspondientes al cuarto año de la carrera. Todo esto además, aplicado sobre un caso real.
- Trabajar en un tema que no fue visto a lo largo de la carrera. La gestión de Riesgos es un tema que no se estudia en ningún curso, por tanto para abordarlo fue necesario investigar desde cero. Este tema además es transversal a varias disciplinas ya que se encuentran trabajos realizados por especialistas en ingeniería, arquitectura, sociología, por mencionar algunas.
- La importancia de ser la primera investigación de gran porte en esta área en nuestro país, según palabras del director del CECOED.

Con respecto al prototipo, se pueden agrupar los resultados positivos en cinco grandes áreas: Tecnologías, Interoperabilidad, Generalidad, Extensibilidad y Facilidad de uso.

Tecnologías:

- Servidor único centralizado:
 - Los datos referentes a los locales (con sus respectivos planos y distribuciones de personas), son almacenados en un servidor de base de datos único. De esta manera se tiene la información centralizada. Los usuarios ingresan al sistema desde un navegador web, que accede al servidor de aplicaciones (arquitectura cliente-servidor).
Esto nos da la ventaja de que los diferentes actores involucrados pueden acceder a la misma información, evitando así los datos duplicados que suelen llevar a conflictos de sincronización.
- Tecnologías de última generación:
 - Para el desarrollo de sistema se utilizó una tecnología de última generación, en crecimiento (Grails), con los riesgos que eso implica. Esta decisión fue tomada ya que se consideró que las ventajas proporcionadas por este Framework, eran más grandes a los riesgos asumidos. Hoy, con los resultados a la vista, se puede confirmar que se obtienen grandes ventajas a la hora de desarrollar y ejecutar la aplicación (fácil creación a partir de las clases de dominio de los archivos que componen la lógica, presentación y comunicación con la BD)
- Software libre:
 - Tanto el manejador de base de datos, el framework de desarrollo, el servidor de aplicaciones, así como los web services consumidos, son en su totalidad de libre uso y distribución. Si bien no era un requerimiento inicial, es un aspecto sumamente positivo del proyecto, dado que entre otros factores, implica costo cero en licencias. Además todas estas herramientas cuentan con reconocimiento y soporte a nivel mundial, lo que garantiza, su estabilidad, y futuras mejoras.

Interoperabilidad:

- Plataforma web:
 - Gracias a que el prototipo fue construido sobre una plataforma web, con una arquitectura cliente-servidor, es posible acceder a la aplicación, únicamente contado con acceso a un navegador web con acceso a la red donde esta alojada la aplicación. Además, cualquier actualización del sistema, será inmediatamente utilizada por todos los clientes.

- Importar/Exportar:
 - Las funcionalidades de Importar y Exportar provistas por el prototipo, permiten que cualquier usuario pueda tener su propio ambiente de simulación (con Locales, Planos, Distribuciones en una BD propia), y distribuirlo al resto de los usuarios, o eventualmente al servidor central.
Esto entre otras cosas, permite que cada entidad (Shopping, Supermercados, Discotecas etc.), puedan mantener de manera actualizada la información referente a su Plano, y sincronizar dicha información con el servidor central, simplemente exportando sus datos.
- Webservices:
 - En la creación de locales, el software utiliza los servicios de Google Maps para obtener la georeferencia (coordenadas) del lugar. La conexión se realiza por medio de un webservice. Este es un aspecto positivo ya que se obtienen datos relevantes desde una fuente externa, en un futuro se podría conectar con otros sistemas remotos para obtener otro tipo de información.
Si bien las coordenadas, simplemente son guardadas como un dato adicional del Local, consideramos importante su inclusión ya que es una puerta abierta para trabajos futuros, tales como el traslado de heridos hacia centros asistenciales.

Generalidad:

- Una de las ventajas más importantes es la generalidad del producto. Es decir, no fue realizado para un local particular, sino que, por el contrario, es posible crear, administrar y trabajar con distintos tipos de locales.

Extensibilidad:

- Del sistema construido:
 - Todo el proceso de análisis, diseño y construcción del sistema, se basó, sobre la idea de extensibilidad.
En tal sentido, por ejemplo, es posible agregar de manera relativamente fácil, nuevos objetos dentro de un piso, (Puertas deslizables, ventanas, etc.), y que dichos objetos, sean tenidos en cuenta, al momento de simular.
- A futuro
 - Si bien el prototipo desarrollado, trata un tipo específico de Aglomeraciones (en Locales cerrados). Los conceptos, utilizados tanto para el diseño del Local, y fundamentalmente, en los algoritmos utilizados para la simulación no se restringen únicamente a este tipo de Aglomeraciones, si no que por el

contrario, pueden ser aplicados sin restricciones a otros tipos de Aglomeraciones.

Facilidad de uso:

- Diseño del ambiente:
 - El sistema nos permite manejar distintos planos para un mismo centro comercial. De esta manera, además de hacer pruebas con los planos actuales, podemos ingresar planos de ensayo para simular y comparar las salidas con el plano actual. Esto fue pensado para manejar alternativas de posibles mejoras edilicias que brinden, por ejemplo, mayor velocidad de evacuación ante un eventual siniestro. De igual manera, es posible trabajar con distintas distribuciones de personas para un centro comercial, que varían según el día (semanal) y hora.
 - Tanto el diseño del ambiente, como la simulación de un siniestro, se visualiza todo bajo la misma interface, esto facilita la comprensión, y utilización del sistema.

En acuerdo con el cliente (CECOED) y los tutores del proyecto, algunos requerimientos o funcionalidades, fueron dejadas fuera del alcance final de la implementación del prototipo, es por eso que los comentaremos como resultados “negativos” del mismo.

Sistema:

- Sin manejo de Usuarios:
 - Actualmente el prototipo desarrollado, no posee restricciones, respecto al uso por parte de los usuarios, sin embargo sería deseable, que las distintas funcionalidades, estuvieran restringidas al tipo de usuario, de esta forma, a modo de ejemplo, un usuario común de la aplicación, podría ver información referente a los Locales, pero no editar su información.

Diseño:

- Interface de construcción del piso
 - La fiel correspondencia entre el Local modelado, y la realidad, es clave para que las salidas obtenidas a partir de la simulación de un siniestro sean consideradas de valor. En tal sentido, consideramos que la construcción del Piso, y sus artefactos, si bien contiene los elementos más importantes como ser Puertas internas, Salidas, Conexiones, etc., puede ser mejorada de manera considerable,

incluso, adaptando el software para que se comuniquen con otras herramientas especializadas en el diseño de planos.

- Restringido a Locales cerrados
 - La restricción de trabajar únicamente con Aglomeraciones en locales cerrados, es un aspecto negativo, del prototipo construido resultado de la definición acotada del alcance.

Simulación:

- Variedad de Salidas:
 - No proveer distintas salidas de acuerdo al usuario que utilice el sistema, es otro resultado negativo derivado del alcance del proyecto. Se podría incluir salidas, para los distintos equipos de emergencia (Médicos, bomberos, personal de seguridad), público en general, etc.

Pruebas:

- Caso Real:
 - Luego de realizado el producto, se llevaron a cabo una gran cantidad de pruebas y algunas demostraciones interesantes. Se observa la utilidad que puede tener, permitiendo administrar y evaluar informaciones valiosas. Se trabajó con locales y planos de distintos tipos y tamaños. Sin embargo, si bien los datos ingresados fueron coherentes y consistentes, los mismos eran ficticios. Sería positivo que el producto sea utilizado con datos reales de algún local particular, y en conjunto con los encargados de la seguridad del local.

8.2 Conclusiones

Desde fines del siglo pasado, el desarrollo urbano ha tenido un gran incremento, a tal punto que se podría catalogar como exponencial.

Dicho incremento poblacional obligó al desarrollo estructural (edificios, casas, comercios, calles, etc.). De esta manera, también se hicieron frecuentes las grandes aglomeraciones de público en centros comerciales, eventos musicales, deportivos, etc.

En las últimas dos décadas han sucedido varios desastres o situaciones de emergencia en lugares o eventos con grandes aglomeraciones de público. Algunos casos nos llevan a reflexionar sobre la falta de prevención, organización y preparación, que es, a veces hasta inexplicable.

Por irresponsabilidad o abuso de unos pocos, ya sea organizadores y/o participantes, suelen ser afectadas muchas personas. Alguno de los lugares en

que ocurrieron desastres no estaban preparados para soportar mucha gente o no fueron debidamente controlados.

Lamentablemente, en general la discusión y cuestionamiento de estos temas se realiza después que suceden los desastres. Por ello, es fundamental cambiar esta mentalidad para comenzar a trabajar en la prevención del problema.

Las aglomeraciones conllevan riesgos, y el interés por su gestión ha aumentado, lo que se refleja en la creación de variadas instituciones y organizaciones, tanto nacionales como regionales e internacionales, dedicadas a la prevención y mitigación de los mismos.

La gestión de aglomeraciones debe tratarse considerando, por un lado, el lugar concreto en que suceden (estadio, shopping, centro comercial, teatro), y, por otro lado, las características del área o zona subyacente (calles, barrio, ciudad).

Las áreas informáticas de Investigación Operativa, Simulación y Sistemas de Información Geográfica son las más afines con estos temas.

En la evacuación del lugar concreto o local, los cálculos matemáticos y el manejo de grafos tienen un papel importante, ya que son necesarios para optimizar la evacuación tomando en cuenta la distribución de las personas, sus velocidades, movimientos, las distancias hacia las salidas y el acceso a la zona siniestrada por parte del personal de emergencia.

Cabe destacar que el tratamiento de aglomeraciones podría involucrar varias personas (responsables y organizadores de los eventos, instituciones de emergencia, informáticos, participantes del evento) que, si bien tienen distintos perfiles, deberían trabajar en conjunto, aportando cada uno lo suyo para analizar y definir planes de contingencia comunes, tanto del local como del área de evacuación.

Actualmente los sistemas computacionales tienen un papel importante debido a su rapidez, precisión y constante crecimiento, haciendo que la gestión de riesgo en las aglomeraciones comience a tener una estrecha relación con dichos sistemas. Es por ello, que sería bueno que los ingenieros o informáticos que elaboran los sistemas (programas) utilizados, tengan conocimiento de la gestión de riesgo en las aglomeraciones. A su vez, los expertos en la gestión de riesgo, deben comprender el funcionamiento de los sistemas informáticos, para utilizarlos y manejarlos adecuadamente.

Gracias al procesamiento computacional se obtienen datos valiosos en poco tiempo, realizando cálculos que a veces requieren mucho tiempo y dedicación obtenerlos por otros medios. De esta manera, los recursos informáticos podrían encargarse de procesar y realizar cálculos complejos, permitiendo a los

especialistas en gestión de riesgo, “ganar” tiempo para definir estrategias a partir de los resultados obtenidos.

En este sentido, es fundamental que tales resultados sean confiables, ya que, a partir de ellos, se tomarán decisiones importantes que ayuden a evitar o disminuir el riesgo de desastre.

La correctitud, y muchas veces la exactitud de los resultados son claves para definir buenas estrategias de prevención, mitigación y preparación ante posibles eventos adversos.

Para tomar decisiones y definir estrategias suelen tener cabida una infinidad de factores. Estos van desde aspectos generales como la ubicación de las puertas, salidas y vías de circulación; hasta los más particulares o específicos como el flujo de las vías, velocidad y edad de las personas. El hecho de considerar una gran cantidad de elementos, se verá reflejado en la mayor calidad y eficacia de las decisiones adoptadas. Sin embargo, cuanto mayor sea la cantidad de factores a tomar en cuenta, también se incrementa la complejidad, y en consecuencia, el tiempo para adoptar resoluciones. Es por ello, que, antes de enfocarse en particularidades, lo ideal es considerar primero los aspectos generales, a partir de los cuales sea posible elaborar estrategias básicas. Luego, se pueden ir incorporando los factores específicos. En el proyecto se usó dicha metodología y esta experiencia nos permite afirmar que es una buena opción de trabajo.

Antes de comenzar el proyecto el equipo de proyecto no contaba con ninguna experiencia práctica ni estudios teóricos sobre temas relacionados a la gestión de riesgos. Sólo se tenía conocimiento de algunos desastres ocurridos en los últimos años, que afectaron a muchas personas y que habían conmocionado al equipo.

Es así, que surgió el interés de aprender e investigar los avances en el área de la cual se tenía poco conocimiento, y principalmente, tratar de contribuir de alguna forma a evitar los desastres.

A medida que se fue abordando el tema en las primeras reuniones con los clientes y tutores, el equipo notó que es un área muy amplia, en la que se está trabajando e intentando mejorar, apostando a la integración de varias ramas de investigación. Gracias al trabajo en equipo junto a clientes y tutores, se lograron avances en la integración de la informática a estos asuntos.

El equipo de proyecto cree que el sistema desarrollado es un aporte valioso a la sociedad entera y se satisface por poder contribuir de alguna forma a la prevención de desastres y mitigación de riesgos.

La gestión de riesgos en general, y en particular, la de los siniestros en locales de grandes aglomeraciones de público, son temas muy amplios, que involucran una gran cantidad de factores; siempre hay cosas para mejorar.

Como se mencionó anteriormente, el trabajo realizado pretende ser un avance más en el tema. No es una versión final, sino que intenta ser un aporte y motivador para seguir trabajando en el área, mejorándolo o iniciando nuevos estudios a partir de él.

Se considera, que el proyecto desarrollado contribuye en cierta forma a lograr el objetivo principal, que es la preservación de vidas humanas.

8.3 Trabajos futuros

Tomando en cuenta por un lado los resultados “negativos” del proyecto, es decir aquellos puntos que por diversas razones quedaron fuera del alcance del proyecto, e incluyendo algunos puntos que surgieron como posibles mejoras a la aplicación en vistas a su futuro desarrollo, elaboramos una lista de mejoras o trabajos futuros, que detallamos a continuación:

- Modulo Diseño:
 - Mejoras en algunas secciones de la interface, por ejemplo en el diseño de los planos. La edición de los planos y las estructuras de sus pisos pueden ser más amigables, para lo cual sería bueno dialogar y analizar propuestas junto con los usuarios finales.
 - Permitir importar planos diseñados en otros programas especializados en esa tarea (por ejemplo, autocad). En un primer análisis e investigación, se vio que no era una tarea sencilla, que podía insumir mucho tiempo y en la que no se puede asegurar su factibilidad. Es por ello y considerando que el proyecto no está centrado en esta funcionalidad, se decidió postergarla para trabajos futuros. En esta etapa sólo se brindó la posibilidad de importar y exportar planos en un formato previamente determinado, específico para el sistema, sin embargo, permitir la interoperabilidad con otros sistemas especializados, aportaría al proyecto de una mayor utilidad.

- Modulo Simulación:
 - En un futuro es recomendable que el sistema incorpore salidas que estén destinadas a otros actores involucrados, como el personal de seguridad o público en general que se encuentre en el local. El hecho de que todos los actores estén informados contribuye a

mejorar la prevención y agilizar la evacuación/rescate en caso de siniestros.

- Otra extensión que se puede realizar, en el módulo de Simulación, es la inclusión de otra salida que muestre como se propagaría el siniestro tomando en cuenta el tipo de siniestro (incendio, derrumbe, etc).
- Sistema en general:
 - El software de Aglomeraciones por cómo fue diseñado permite la incorporación de un módulo de manejo de usuarios para definir distintos tipos de usuarios con diferentes privilegios, por ejemplo, que un usuario común cree el local y un usuario administrador sea el que pueda crear el mapa del local.

La emergencia de un siniestro en un evento de gran aglomeración de público amerita que se cuente con planes de respuesta referidos a la contención del siniestro, evacuación de la población, atención de los heridos y accesibilidad de los vehículos de emergencia. Como trabajo a futuro se estima conveniente que se realice el acceso de los vehículos de emergencia y traslado de evacuados hacia los centros asistenciales mas adecuados.

Como se mencionó en el alcance del proyecto, el sistema generado es un prototipo. Se pretende que el mismo sea inicio y motivador para la creación aplicaciones más completas y mejoradas, de acuerdo a las exigencias y necesidades de los usuarios finales.

Sabido es que en el mundo actual, lo constante es el cambio. Continuamente surgen nuevas y mejores tecnologías, así como estrategias a utilizar para la prevención, mitigación y respuesta ante siniestros. Es deseable que los futuros programas estén actualizados, acompañando los cambios en ambos sentidos. De esta manera los programas estarían en una mejora continua.

9 Manual de Usuario

9.1 Página de Inicio

Aquí se despliega la página de inicio que se muestra más abajo, con los dos módulos principales del sistema, el módulo de Diseño y el módulo de Simulación.

Para volver desde cualquier página a la página de inicio, se debe clickear en el menú de la izquierda en "**inicio**".

En el módulo de Diseño, se puede acceder a las funcionalidades de creación de los Locales, así como también las distribuciones de personas en cada uno de estos.

En el módulo de Simulación, se muestran las salidas del sistema que se deseen simular, como por ejemplo la ruta óptima de evacuación, al ocurrir un siniestro en determinado local con cierta distribución.



Figura 18 Pantalla de Inicio

9.2 Módulo Diseño

9.2.1 Crear Local

Al ingresar al módulo de diseño, lo primero que se observa es la lista de Locales existentes.

Dentro de esta vista, tenemos tres funcionalidades:

- Acceder a un Local, seleccionando el Local desde la lista
- Crear Local
- Importar Local

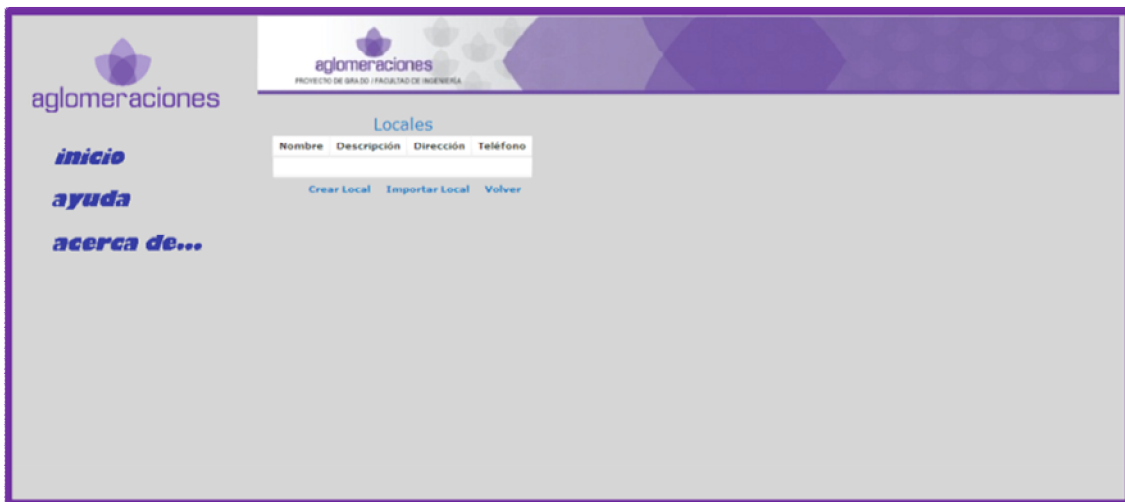


Figura 19 Módulo Diseño

Al realizar click en “Crear Local” se abre una pantalla (ver figura 20) para ingresar los datos del local, como ser Nombre, Descripción, Dirección y Teléfono, al confirmar los datos se muestra un mapa donde se debe elegir la ubicación del Local creado.

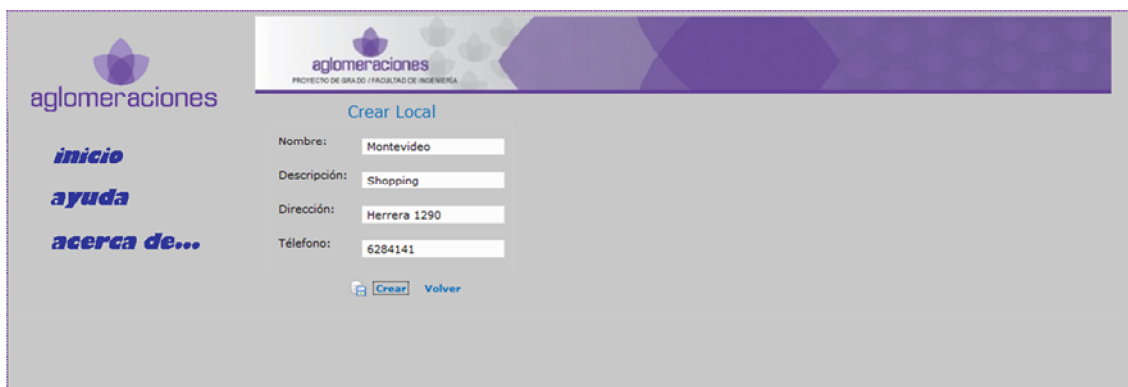


Figura 20 Crear Local – Datos Básicos

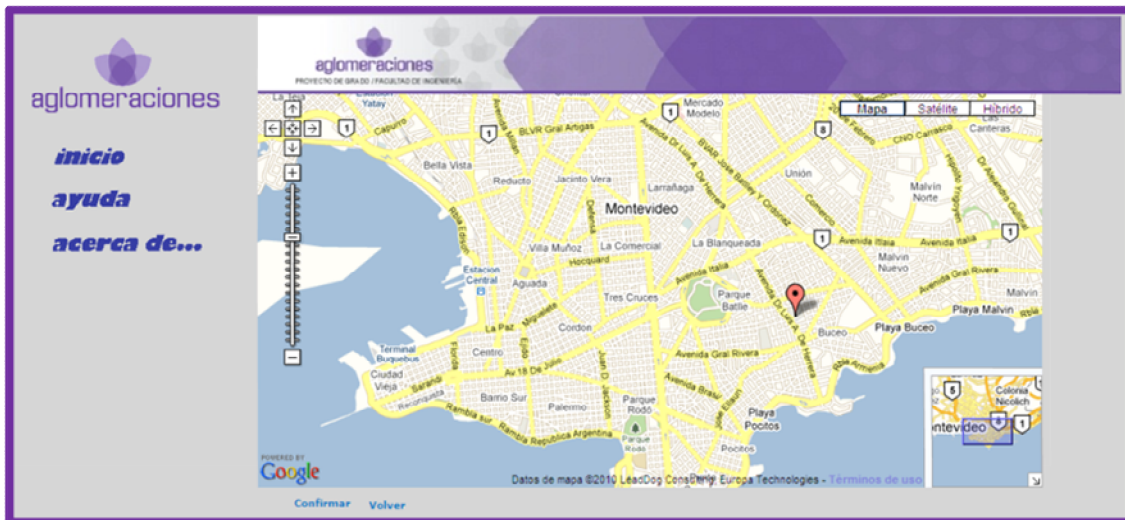


Figura 21 Crear Local – Ubicación

9.2.2 Trabajar con Local

Al momento de trabajar con un Local podemos:

- Editar: con ello, accedemos a los datos del Local y los podemos modificar.
- Exportar: mediante lo cual, se genera un archivo, con extensión .agl, con toda la información asociada al Local, de manera de poder transferir estos datos a otro sistema.
- Importar Plano: donde, a partir de un archivo, con extensión .agl, podemos, importar los datos de un Plano.
- Borrar: el cual, borra (previa confirmación) toda la información asociada a un Local.
- Agregar Plano: esta funcionalidad se describe con profundidad en la siguiente sección.

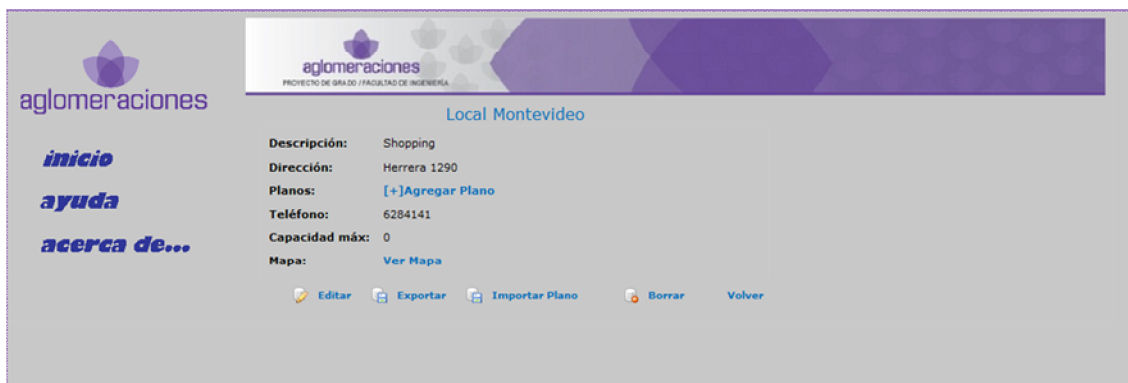


Figura 22 Trabajar con Local

9.2.3 Crear Plano

En la figura 23 se muestran los datos que se solicitan para crear un plano. Al presionar en "Crear", se habilitará una página (ver debajo) desde donde se pueden agregar los pisos, las conexiones (escaleras, ascensores, rampas) y las distribuciones de personas al plano.

También es necesario indicar la escala del plano con la cual se creará.

The screenshot shows a web interface for creating a plan. On the left, there is a sidebar with the 'aglomeraciones' logo and navigation links: 'inicio', 'ayuda', and 'acerca de...'. The main content area is titled 'Crear Plano'. It contains the following form fields:

- Nombre:** Actual
- Ancho:** 25
- Largo:** 25
- Local:** Montevideo
- Escala del Plano:** 1:25 (dropdown menu)

At the bottom of the form, there are two buttons: 'Crear' and 'Cancelar'.

Figura 23 Crear Plano

The screenshot shows the 'Trabajar con Plano' (Work with Plan) interface. The sidebar is the same as in Figure 23. The main content area is titled 'Local Montevideo | Plano Actual'. It displays the following parameters and actions:

- Ancho:** 25
- Largo:** 25
- Pisos:** [+] Agregar Piso
- Conexiones Entre Pisos:** [+] Agregar Conexión
- Distribuciones:** [+] Agregar Distribución

At the bottom, there are four action buttons: 'Editar', 'Exportar', 'Borrar', and 'Volver'.

Figura 24 Trabajar con Plano

9.2.4 Crear Piso

Para crear un piso asociado a un plano, se debe ingresar primeramente el número de piso que se está creando y presionar en "Crear".

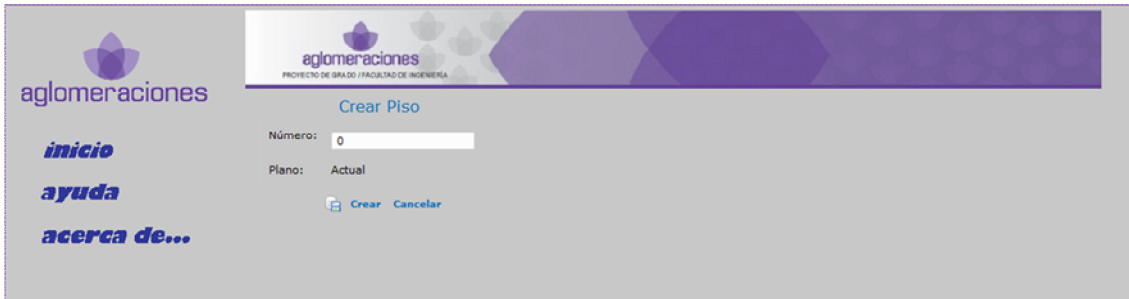


Figura 25 Crear Piso

Como se muestra debajo aparece una cuadrícula que simula ser el piso del plano, allí se debe agregar los componentes del piso como ser paredes, salidas, etc.

Para agregar una pared por ejemplo, se debe seleccionar "Pared" en "Pintar", y con el botón izquierdo del mouse se debe pintar sobre la cuadrícula.

Para eliminar las estructuras del piso se selecciona "Vacio" en "Pintar" y se pinta en la cuadrícula sobre la estructura que se desee quitar.

Para culminar la creación del piso se debe hacer click en "Guardar".

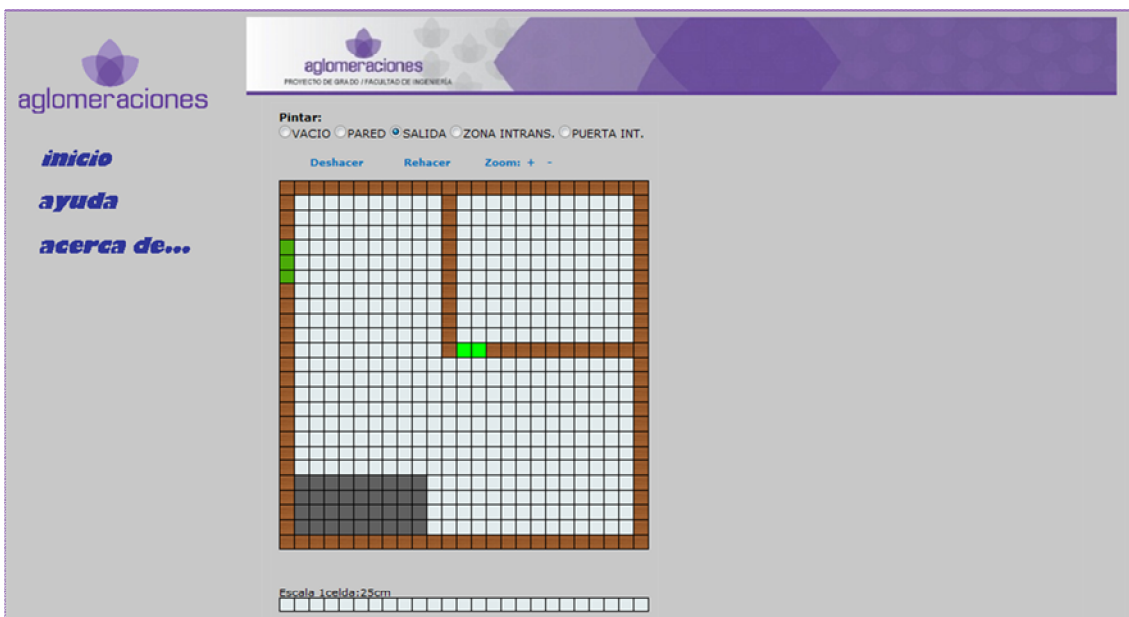


Figura 26 Editar Piso

9.2.5 Crear Conexión

Para poder agregar una conexión al plano se deben tener al menos dos pisos creados.

Se debe seleccionar: el tipo de conexión a crear, los pisos que conectará y en el caso de crear una escalera o rampa también se ingresarán las medidas de la conexión.

Para culminar la creación de la conexión al clickear en "Guardar", se habilitan las páginas con la cuadrícula del piso inferior y superior para seleccionar el lugar donde se colocará la conexión.

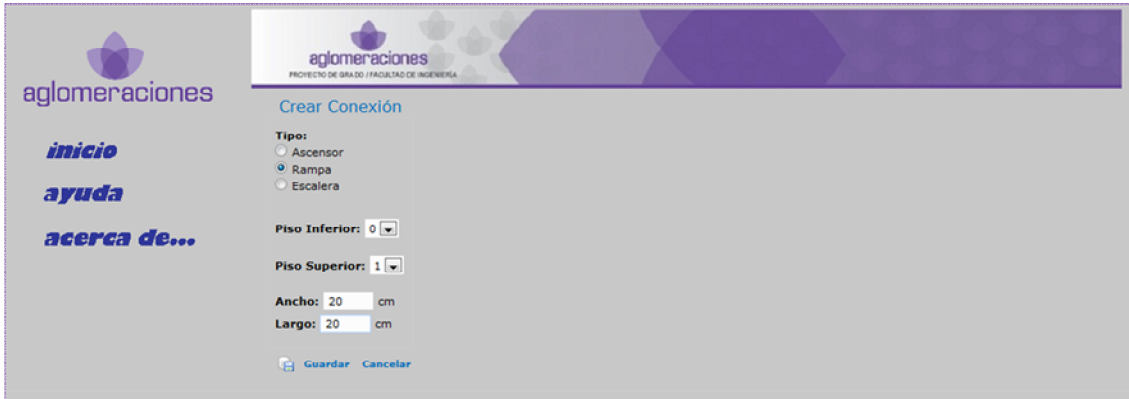


Figura 27 Crear Conexión - Datos

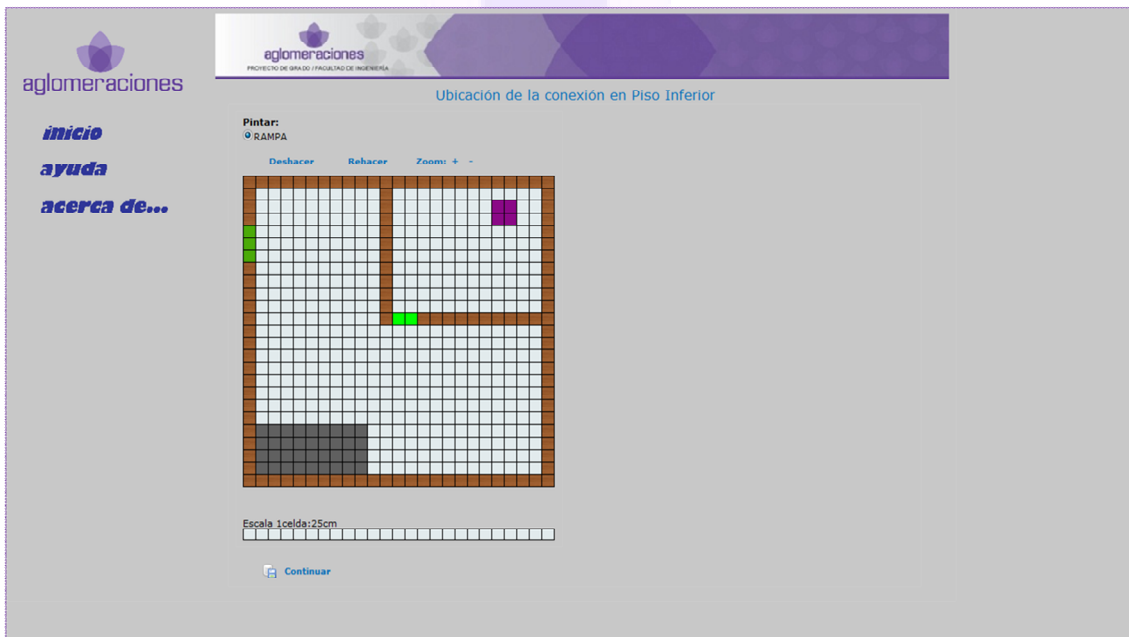


Figura 28 Crear Conexión – Ubicación Piso Inferior

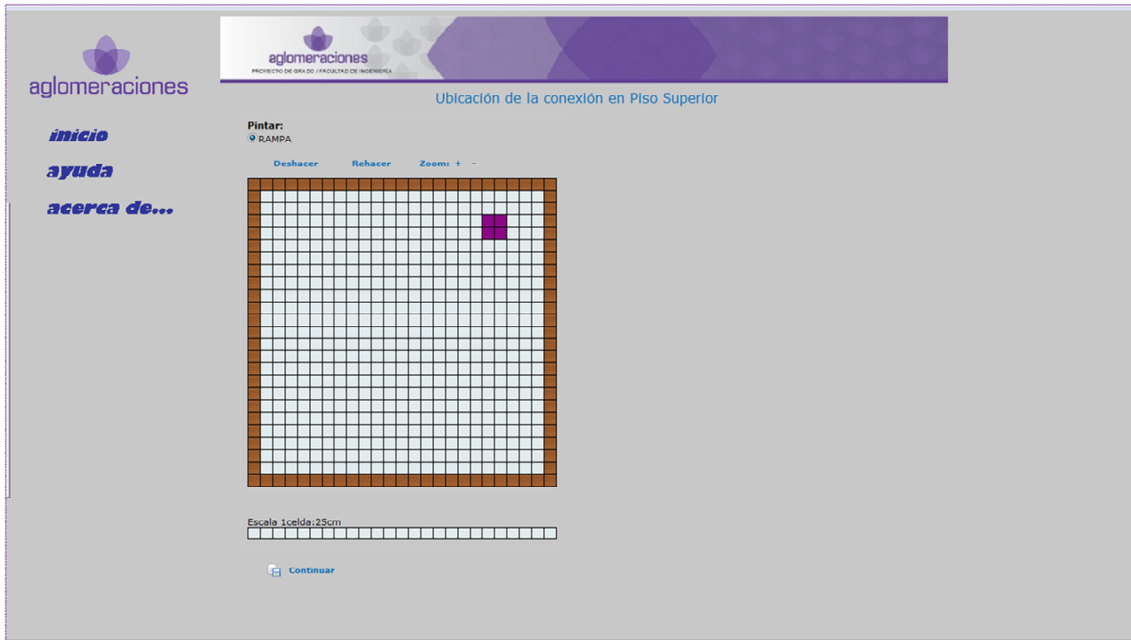


Figura 29 Crear Conexión – Ubicación Piso Superior

9.2.6 Crear Distribución

Para poder crear una distribución de personas, se deben ingresar descripción de la misma, y a que periodo de la semana hace referencia, y luego presionar “Guardar”.

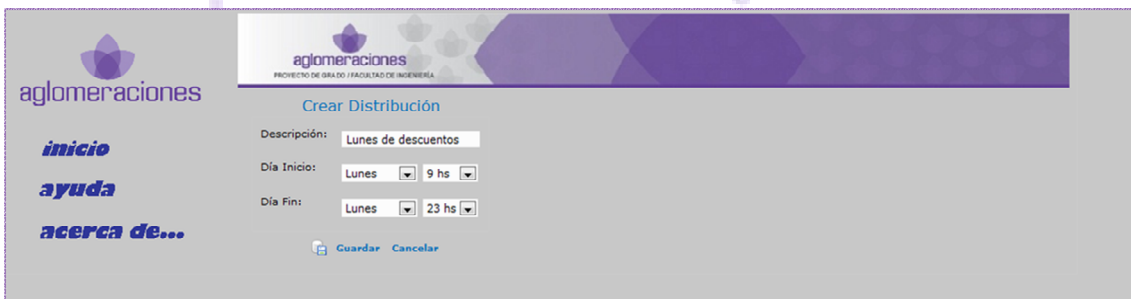


Figura 30 Crear Distribución

Se habilita la siguiente página donde se podrá elegir el piso para ubicar a la distribución de personas.



Figura 31 Editar Distribución

Se debe hacer click en el piso correspondiente para poder agregar a las personas, pintando en la cuadrícula como se hizo con las estructuras del piso.

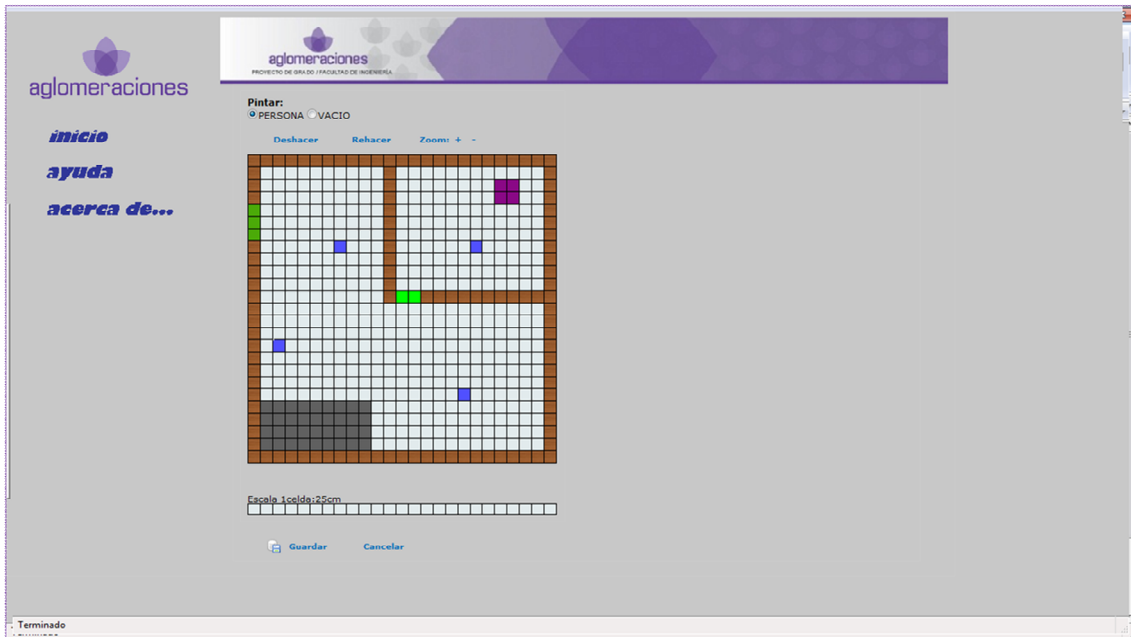


Figura 32 Agregar Personas (Piso 0)

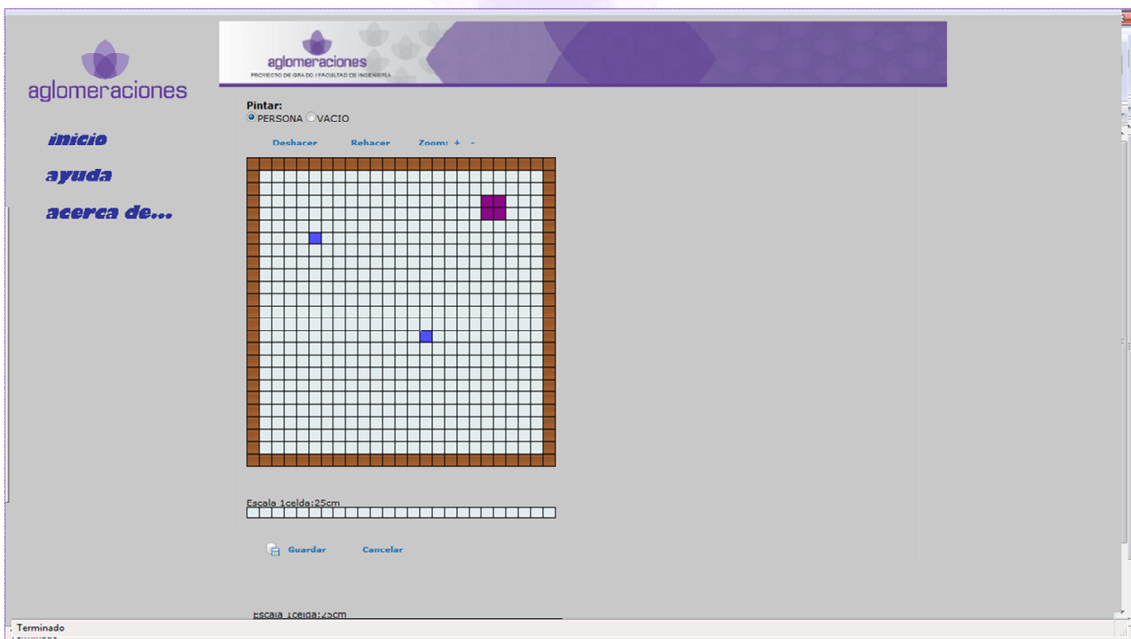


Figura 33 Agregar Personas (Piso 1)

9.3 Módulo Simulación

Desde este módulo se simulan las distintas salidas para los locales ya creados en el módulo de Diseño.

9.3.1 Pasos para Simular

9.3.1.1 Selección del Local

En la pantalla que se muestra debajo están los locales que se crearon en Diseño, se debe seleccionar el local sobre el cual se desea realizar la simulación.



Figura 34 Selección del Local

9.3.1.2 Selección del Plano

A partir del Local seleccionado, se indica, con cual Plano asociado al Local se desea trabajar.



Figura 35 Selección del Plano

9.3.1.3 Selección de la Distribución

Se indica, con que distribución de personas, asociadas al plano se va a realizar la simulación.



Figura 36 Selección de la Distribución

9.3.1.4 Selección del Piso Siniestrado

En este paso, se selecciona, el piso dentro del plano, donde se produce el siniestro.



Figura 37 Selección del Piso

9.3.1.5 Ubicación del Siniestro

Luego de elegir el piso del siniestro, se despliega la página siguiente con la cuadrícula del piso en donde se delimita el tamaño y la ubicación del siniestro (pintado con rojo).

Respecto al siniestro se debe de indicar:

- El nivel de propagación, o sea a partir de la ubicación del siniestro, cual es la propagación inicial del mismo
- La intensidad, es decir, cuanto se ven afectadas las zonas cercanas al siniestro.
- La velocidad de propagación, este parámetro indica cuanto (en metros por segundos) se propaga el siniestro.

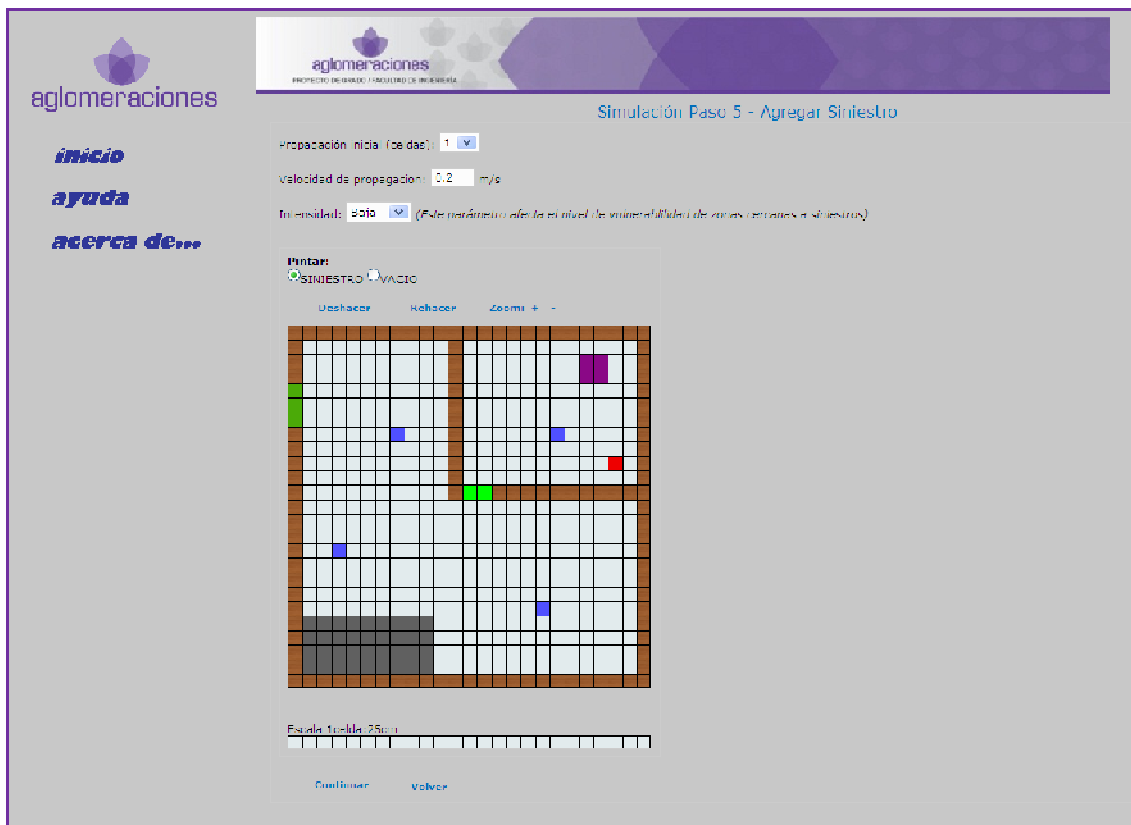


Figura 38 Ubicación del Siniestro

9.3.1.6 Elección de las salidas

En el último paso de la simulación, se debe elegir cuál es la salida deseada. La cuadrícula con la salida se muestra en una ventana popup, por lo que se puede seguir simulando, seleccionando las otras salidas.



Figura 39 Elección de la Salida

9.3.1.6.1 Ruta Óptima de Evacuación

Al clickear en “Ruta óptima de evacuación”, se despliega la página de abajo con las cuadrículas de los pisos del local seleccionado.

Las rutas óptimas de evacuación son los caminos que pueden tomar las personas para salir del local, al suceder un siniestro.

Para obtener la ruta óptima para un grupo determinado de personas se debe seleccionar el grupo de personas en la cuadrícula y luego presionar “Marcar Ruta Óptima”. Modificando la velocidad de desplazamiento, se obtiene para dicho grupo de personas, la distancia y tiempo necesario para salir del Local.

En esta salida, es posible simular la expansión de un siniestro, indicando el nivel de actualización de la simulación. En la simulación, además de observar la expansión del siniestro, se observa el recálculo de rutas óptimas para todos los grupos de personas existentes en la distribución con la que se esté trabajando.

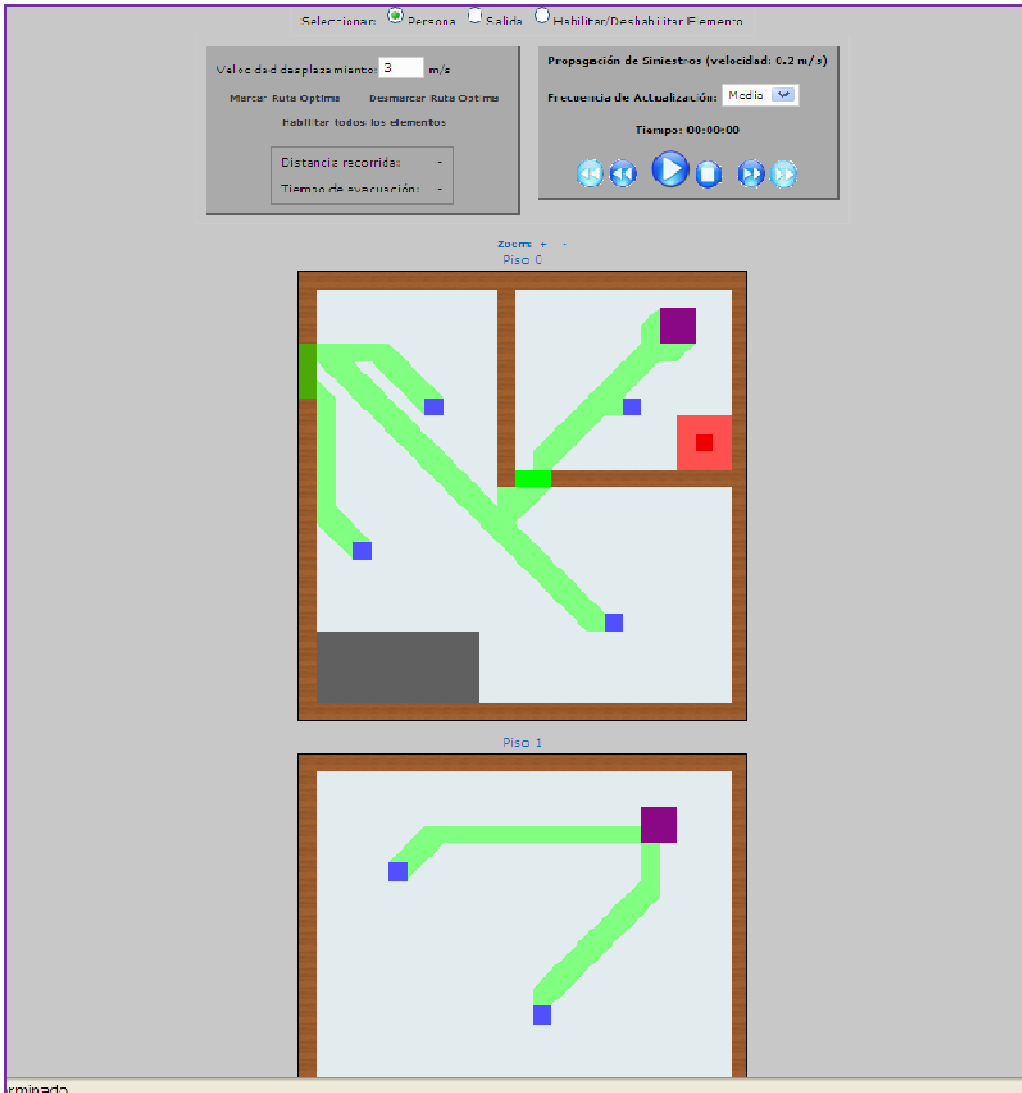


Figura 40 Ruta Óptima de Evacuación

Lo que se muestra a continuación es la salida del “Marcar Ruta Óptima” para determinado grupo de personas (pintado de azul oscuro). También se despliega la distancia recorrida y el tiempo de evacuación.

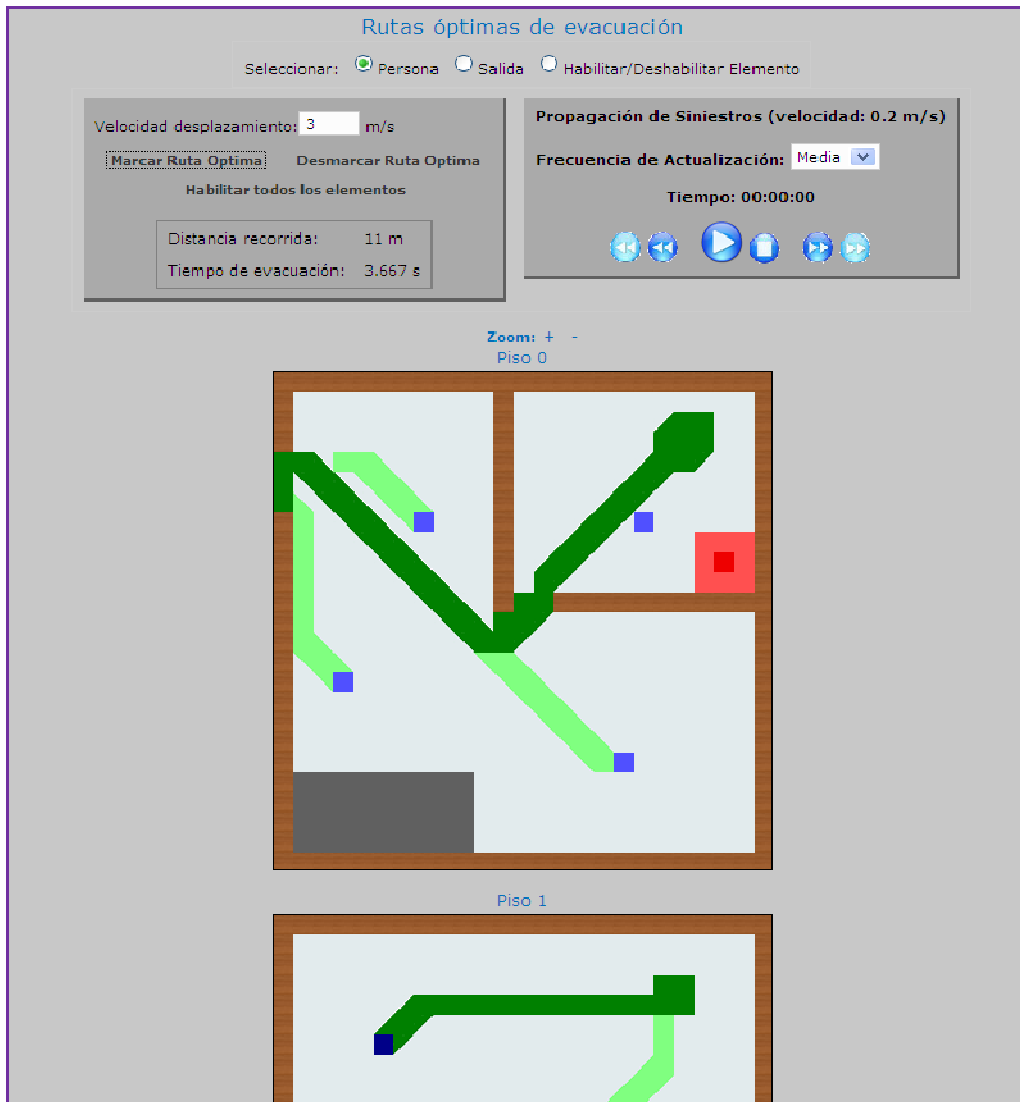


Figura 41 Ruta Óptima de Evacuación - Marcar Ruta Optima

Con la opción "Habilitar/Deshabilitar Elemento", se podría simular la ocurrencia de otro siniestro por ejemplo en la salida del piso 0, como muestra la figura 42, allí se puede notar cómo cambia la ruta óptima de evacuación.

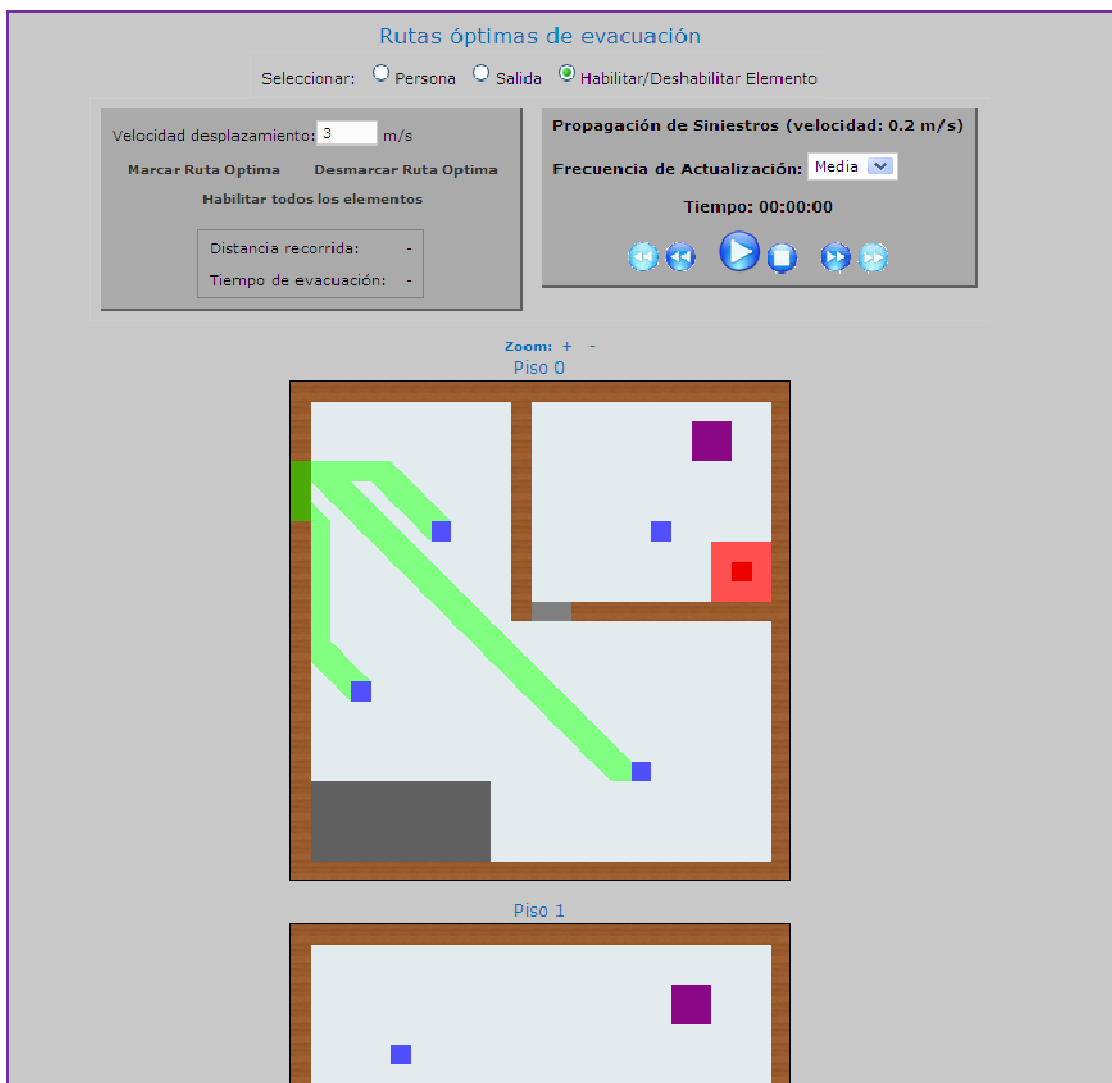


Figura 42 Ruta Óptima de Evacuación - Deshabilitar Salida

9.3.1.6.2 Ruta Óptima hacia el Siniestro

Si la salida seleccionada fuera "Ruta óptima hacia el siniestro", lo que se muestran (ver debajo) son los caminos desde las salidas del local hacia el área siniestrada, esta salida resulta de gran ayuda para el personal de emergencia como ser bomberos, médicos.

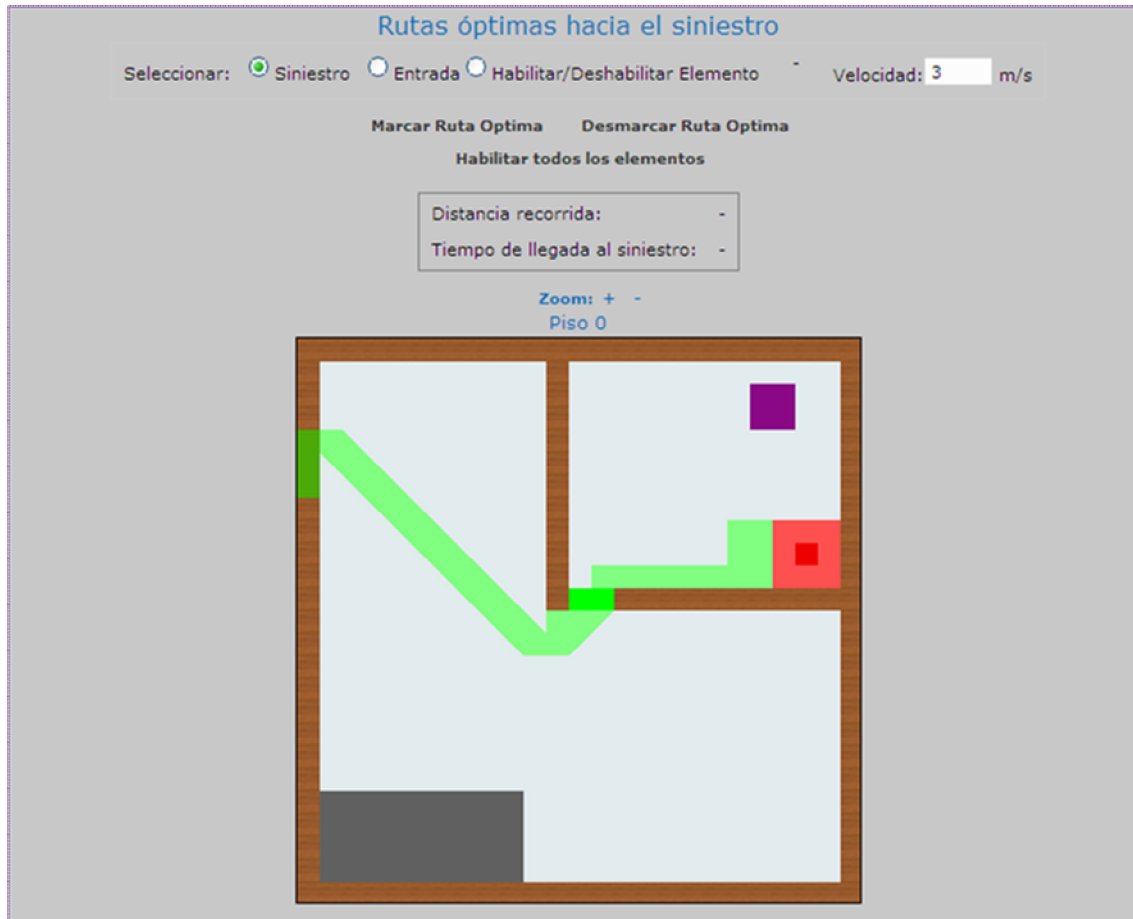


Figura 43 Ruta Óptima hacia el Siniestro

Aquí también se puede seleccionar el "Marcar ruta óptima".

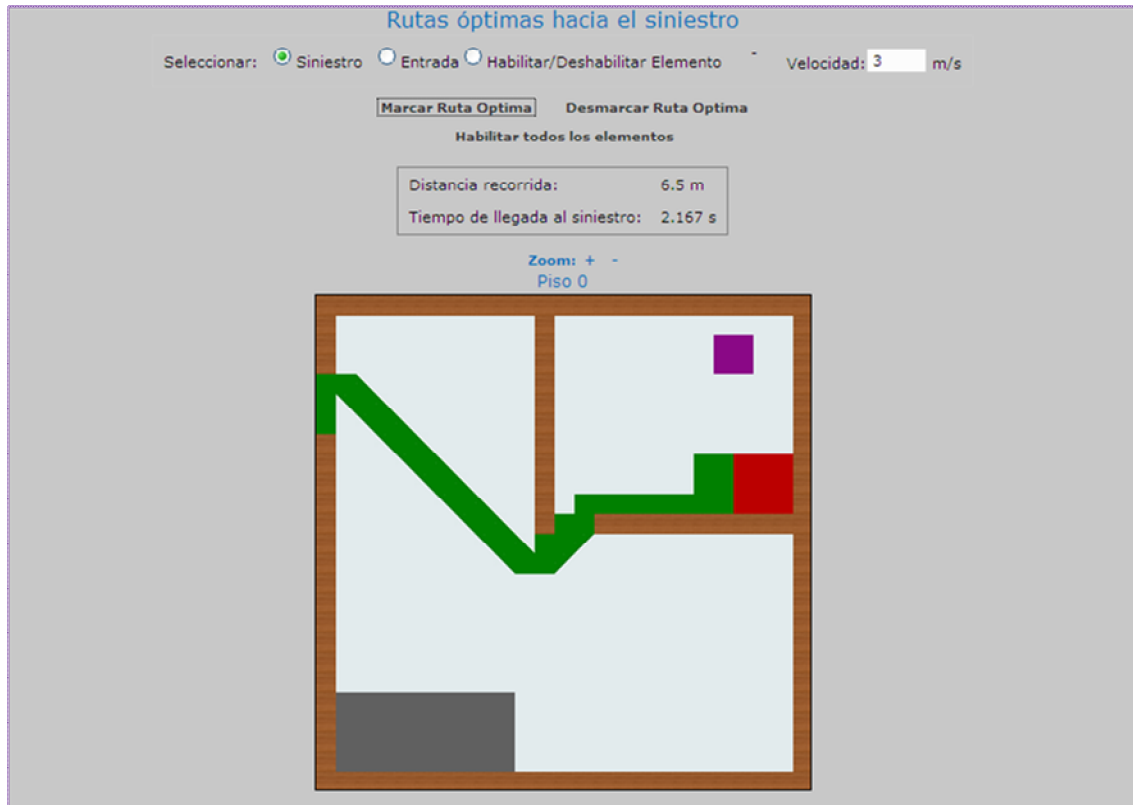


Figura 44 Ruta Óptima hacia el Siniestro - Marcar Ruta Optima

9.3.1.6.3 Niveles de vulnerabilidad

Si se eligiera la salida “Niveles de vulnerabilidad”, lo que se muestra en las cuadrículas de los Pisos (ver debajo), son las áreas más críticas (de rosado a verde), es decir, para cada zona se indica de acuerdo a la distancia a la salida y a los siniestros, la vulnerabilidad de dicha zona.

Es posible simular la expansión de un Siniestro en el tiempo, aquí de acuerdo a la propagación inicial, la intensidad del siniestro, y la velocidad del mismo se puede visualizar la expansión del siniestro en el tiempo, y como ésta afecta las zonas vulnerables.

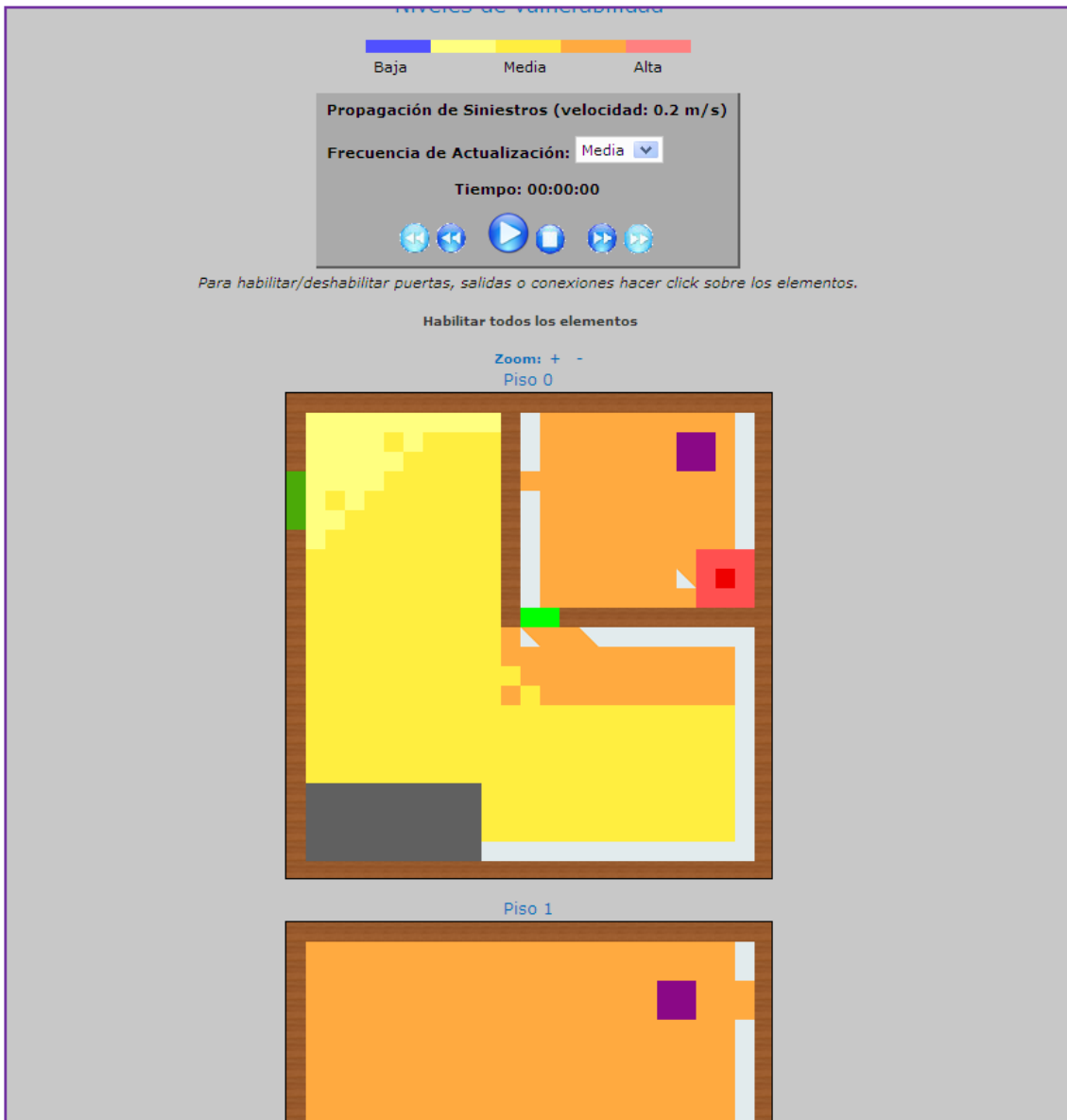
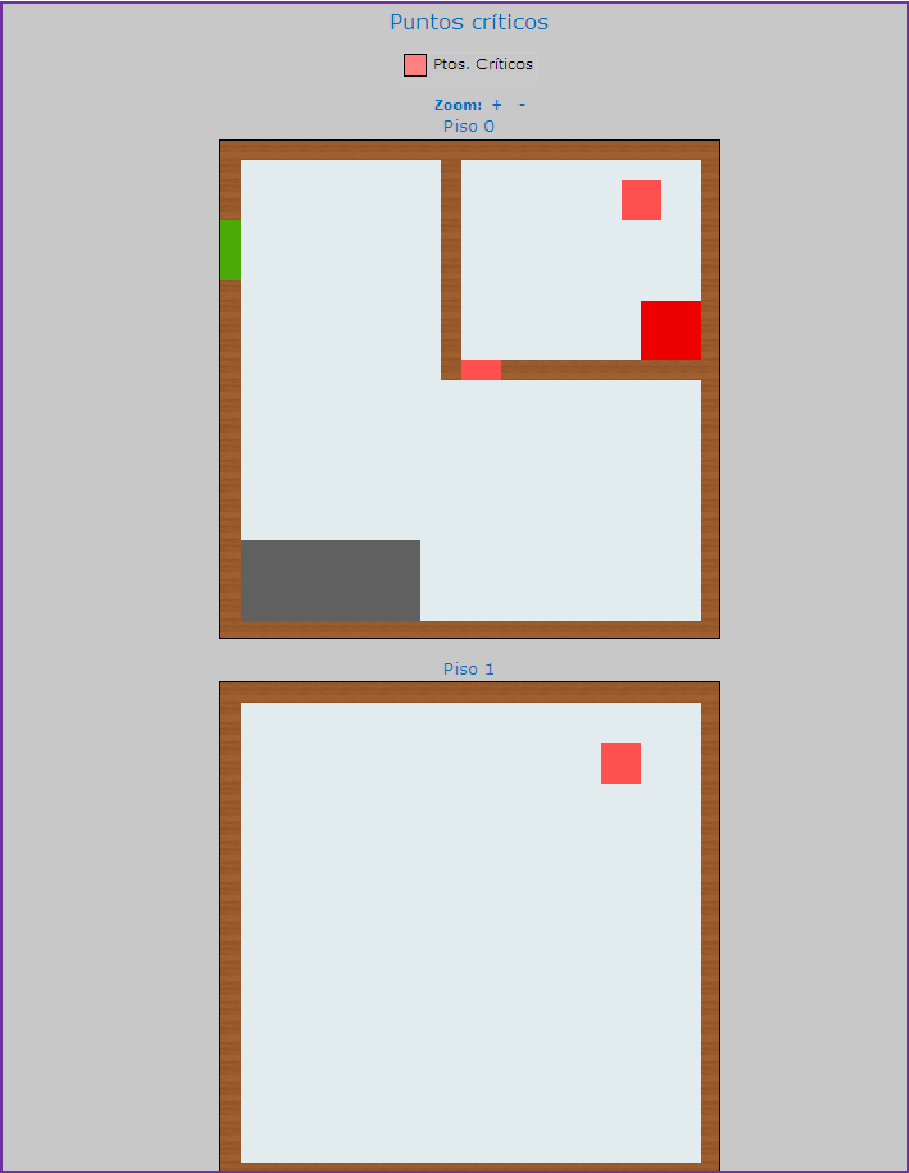


Figura 45 Niveles de Vulnerabilidad

9.3.1.6.4 Puntos Críticos

Al clicar en "Puntos Críticos", se pintarán en las cuadrículas de los pisos del local, las estructuras que de no existir (o estando inhabilitadas por fuego por ejemplo) dejarían desconectadas (sin acceso para el personal de emergencia) ciertas áreas del local.



es

Figura 46 Puntos Críticos

10 Referencias Bibliográficas

[1]

Título: www.180.com

Link: <http://www.180.com.uy/articulo/Falta-de-seguridad-y-desorganizacion-absoluta>

Acceso: 9 de abril del 2010

[2]

Título: TRAGEDIA DEL "YCUA BOLAÑOS"

Link: <http://www.ycuanuncamas.org/www/> y
<http://www.contraincendioonline.com/bolanos.php3>

Acceso: 6 de abril del 2010

[3]

Título: CIDBIMENA

Link: <http://cidbimena.desastres.hn/>

Acceso: 8 de junio del 2009

[4]

Título: Sistema nacional de Emergencias

Link: <http://www.sne.gub.uy/>

Acceso: 7 de mayo del 2010

[5]

Título: Algoritmo de Dijkstra

Link:

http://www.fing.edu.uy/inco/cursos/io/archivos/teorico/cl2y3grafos_07.pdf

Acceso: 18 de mayo del 2009

[6]

Título: Algoritmo búsqueda de puntos críticos

Link: http://www.frikis.org/varios/universidad/pr_08_2005.pdf

Acceso: 18 de mayo del 2009

[7]

Título: EL ABC de las Aglomeraciones - Colombia

Link:

<http://www.fopae.gov.co/portal/page/portal/fopae/eventos/Documentos/EL%20ABC%20DE%20LAS%20AGLOMERACIONES.pdf>

Acceso: 18 de mayo del 2009

[8]

Título: Predes (Centro de Estudios y Prevención de Desastres) - Perú

Link: <http://www.predes.org.pe/>

Acceso: 8 de junio del 2009

[9]

Título: WordReference.com

Link: <http://www.wordreference.com/es/>

Acceso: 8 de junio del 2009

[10]

Título: Grails

Link: <http://www.grails.org>

Acceso: 9 de agosto del 2009

[11]

Título: Groovy

Link: <http://groovy.codehaus.org/>

Acceso: 9 de abril del 2010

[12]

Título: Derby

Link: <http://db.apache.org/derby/>

Acceso: 9 de agosto del 2009

[13]

Título: Tomcat

Link: <http://tomcat.apache.org/>

Acceso: 9 de agosto del 2009

[14]

Título: MySQL

Link: <http://www.mysql.com/>

Acceso: 9 de agosto del 2009

[15]

Título: Google Maps

Link: <http://maps.google.es/>

Acceso: 15 de octubre del 2009

[16]

Título: Asynchronous JavaScript Technology and XML (AJAX)

Link: <http://java.sun.com/developer/technicalArticles/J2EE/AJAX/>

Acceso: 21 de febrero del 2010



11 Anexos

En esta sección se referencian los documentos entregados durante el transcurso del proyecto, y que se anexan al informe final.

- Estado del Arte General

Este documento pretende dar una visión global acerca de la gestión del riesgo, abarcando y profundizando los conceptos asociados al mismo.

- Estado del Arte Aglomeraciones

Este documento da una introducción a la temática del proyecto, enfocado en los siniestros en locales de gran aglomeración de público.

- Documento de Requerimientos

Este documento es especificar los requerimientos del sistema. Los requerimientos surgen a partir de las reuniones mantenidas con el cliente, y el estudio de la realidad en cuestión.

- Documento de Análisis

Se proporciona una visión general del proyecto, los posibles alcances del sistema a construir, y se realiza una justificación del alcance elegido para el proyecto.

- Modelo de Dominio

El informe del Modelo de Dominio ilustra los conceptos del dominio identificados y sus relaciones, además de las restricciones de integridad que aplican sobre ellos. Incluye, además, información (parcial) acerca de los conceptos, los tipos de datos y las relaciones.

- Documento de Casos de Uso

Se especifican los casos de uso del sistema. Los mismos surgen a partir de los requerimientos especificados en el documento "Documento de Requerimientos 2.6.doc".

- Documento de Arquitectura

El documento tiene como cometido especificar formalmente el diseño del sistema, esto incluye su descomposición en subsistemas o módulos, la interacción entre ellos, identificar sus interfaces, realizar el diseño de los casos de uso, de clases y de subsistemas.

- Plan de Pruebas

Se presenta el plan de pruebas funcionales utilizado para testear el proyecto.

