

# Tabla de Contenido

|  |           |
|--|-----------|
| <b>1. Introducción</b>   | <b>1</b>  |
| 1.1. Motivación . . . . .  | 1         |
| 1.2. Objetivos . . . . .   | 2         |
| 1.2.1. Objetivo general . . . . .                                | 2         |
| 1.2.2. Objetivos específicos . . . . .                           | 2         |
| 1.3. Estructura de la memoria . . . . .                          | 2         |
| <b>2. Marco teórico y estado del arte</b>                        | <b>3</b>  |
| 2.1. Conceptos técnicos . . . . .                                | 3         |
| 2.1.1. Física en un choque de cuerpos . . . . .                  | 3         |
| 2.1.2. Sistemas de referencia: posición y rotación . . . . .     | 4         |
| 2.1.3. Procesamiento de señales . . . . .                        | 4         |
| 2.1.4. Histéresis . . . . .                                      | 6         |
| 2.2. Descripción de la empresa y plataforma de trabajo . . . . . | 7         |
| 2.2.1. Empresa Sosmart Labs . . . . .                            | 7         |
| 2.2.2. Hardware . . . . .  | 10        |
| 2.2.3. Software . . . . .  | 15        |
| 2.3. Estado del arte . . . . .                                   | 17        |
| 2.3.1. Soluciones comerciales . . . . .                          | 17        |
| 2.3.2. Publicaciones académicas . . . . .                        | 20        |
| <b>3. Metodología</b>  | <b>25</b> |
| <b>4. Diseño</b>   | <b>26</b> |
| 4.1. Arquitectura del sistema . . . . .                          | 26        |
| 4.2. Algoritmo de detección de choques . . . . .                 | 27        |
| 4.3. Diseño del dispositivo . . . . .                            | 30        |
| <b>5. Implementación</b>   | <b>35</b> |
| 5.1. Algoritmo de detección de choque . . . . .                  | 35        |
| 5.2. Implementación del dispositivo . . . . .                    | 37        |
| 5.2.1. Software Arduino Pro Mini . . . . .                       | 37        |
| 5.2.2. Integración del FONIA 2G . . . . .                        | 38        |
| 5.2.3. Integración del acelerómetro . . . . .                    | 40        |
| 5.2.4. Integración de la tarjeta SD . . . . .                    | 42        |
| 5.2.5. Alimentación del dispositivo . . . . .                    | 43        |
| 5.2.6. Prototipo embebido . . . . .                              | 44        |

|           |  |           |
|-----------|--|-----------|
| 5.3.      | Sistema de atención de emergencia . . . . .  | 45        |
| 5.3.1.    | Base de datos: Back4App . . . . .  | 46        |
| 5.3.2.    | Server web: Recibir POST Request . . . . .   | 47        |
| 5.3.3.    | Dashboard de monitoreo . . . . .   | 47        |
| 5.4.      | Post-procesamiento . . . . .   | 48        |
| 5.4.1.    | Procesamiento de datos de acelerómetro y giróscopo . . . . .                                     | 48        |
| 5.4.2.    | Simulación del choque . . . . .  | 54        |
| <b>6.</b> | <b>Resultados y Análisis</b>   | <b>55</b> |
| 6.1.      | Detección . . . . .  | 55        |
| 6.1.1.    | Pruebas de choques a baja escala . . . . .   | 55        |
| 6.1.2.    | Pruebas de manejo real . . . . .   | 56        |
| 6.1.3.    | Curvas reales de choques de una base de datos . . . . .  | 57        |
| 6.2.      | Notificación . . . . .   | 58        |
| 6.3.      | Post-procesamiento . . . . .   | 58        |
| 6.3.1.    | Filtro de señales . . . . .  | 59        |
| 6.3.2.    | Simulación en Unity3D . . . . .  | 60        |
| <b>7.</b> | <b>Conclusiones y trabajo futuro</b>   | <b>62</b> |
|           | <b>Bibliografía</b>  | <b>64</b> |
| <b>8.</b> | <b>Anexo</b>   | <b>69</b> |
| 8.1.      | Código en Arduino para utilizar FONA 2G . . . . .  | 69        |
| 8.2.      | Código en Ruby On Rails de una sección de la plataforma Web que recibe el POST Request . . . . . | 72        |
| 8.3.      | Código Arduino-MPU9250 . . . . .   | 73        |
| 8.3.1.    | Código adaptado en Arduino para lectura de datos de MPU9250 . . . . .                            | 73        |
| 8.3.2.    | Código en Arduino para el algoritmo de detección . . . . .                                       | 76        |
| 8.4.      | Código test en Arduino para escritura en micro SD . . . . .                                      | 79        |
| 8.5.      | Código MATLAB para procesar señales de aceleración y giróscopo . . . . .                         | 80        |
| 8.6.      | Código C Sharp para simular desplazamientos en Unity3D . . . . .                                 | 83        |
| 8.6.1.    | Código principal para trasladar y rotar . . . . .  | 83        |
| 8.6.2.    | Código para posicionar cámara según movimiento del auto . . . . .                                | 84        |