

1. Pendahuluan

1.1 Latar Belakang

Safety adalah atribut kualitas dari perangkat lunak. Pentingnya keselamatan mengarah pada peningkatan persyaratan *safety* perangkat lunak, yang berfungsi sebagai dasar untuk produk perangkat lunak yang dikembangkan [1]. *Safety requirements* adalah dasar yang ditentukan yang bertujuan untuk meminimalkan risiko dengan membuat prediksi kesalahan, yang mungkin terjadi sebelum peluncuran produk [2]. *Safety requirements* perangkat lunak yang wajar dapat ditunjukkan oleh seberapa banyak pengguna menerima kasus berbeda dari kasus yang diidentifikasi [3].

Salah satu cara untuk mengidentifikasi *safety requirements* adalah dengan menganalisis karakteristik *safety* yang dijelaskan dalam dokumentasi yang ada [4]. Analisis dilakukan dengan beberapa aspek, antara lain fungsional, kegagalan sistem, analisis dampak, dan tindakan pencegahan untuk setiap kesalahan dan kesalahan [5]. Beberapa metode saat ini diperkenalkan untuk menganalisis *safety* perangkat lunak, seperti pohon kesalahan, HazOp, dan FMEA dengan masing-masing metode memiliki kekuatan dan kelemahan yang berbeda [6]. Namun, metode tersebut tidak menyoroti *safety* perangkat lunak dari sudut pandang pengguna, yang penting dalam merancang tindakan pencegahan [6].

Untuk melihat sudut pandang pengguna dapat dengan memanfaatkan *misuse cases*, metode analisis *safety* yang diperluas dari model *use case* [6]. *Misuse cases* menunjukkan serangkaian tindakan yang tidak diinginkan selama penggunaan produk perangkat lunak dari sudut pandang pengguna yang berbeda [7]. Sebelumnya digunakan untuk menganalisis ancaman *security*, *misuse cases* telah diadopsi untuk mengidentifikasi ancaman internal dari kesalahan pengguna dan kegagalan sistem [8]. Implementasi *misuse cases* itu sendiri berisi beberapa langkah, mulai dari mengurai kebutuhan yang ada, mengembangkan diagram UML, menganalisis potensi kegagalan, dan terakhir melakukan validasi hingga menyelesaikan model *misuse cases* yang dikembangkan [6].

Guna mendukung proses analisis celah *safety* pada suatu produk, dibutuhkan program yang dapat melakukan proses pengembangan model diagram dari setiap kasus yang ada. Penggunaan program ini bertujuan untuk mempersingkat waktu yang dibutuhkan untuk menghasilkan suatu model. Program yang dikembangkan akan menghasilkan model diagram berdasarkan kata/aksi yang dimasukkan. Untuk menghasilkan model diagram dari *misuse cases* terdiri dari beberapa langkah, mulai dari proses *parsing* dari setiap kata yang dimasukkan, pencocokan kata dengan kamus data yang sudah ditentukan, dan terakhir setiap kata yang ada akan ditentukan actor, relasi, dan juga entitas yang dilakukan.

1.2 Topik dan Batasannya

Penelitian ini akan berfokus pada kapabilitas *misuse cases* dalam pemodelan *safety requirements* perangkat lunak. Penerapan metode *misuse cases* digunakan untuk menganalisis *safety* sistem perangkat lunak autopilot mobil listrik. Pemilihan kasus didasarkan pada data yang menunjukkan risiko kritis dari keselamatan pengemudi ketika sistem autopilot gagal dalam skenario lingkungan yang sebenarnya [9, 10]. Penelitian ini akan fokus pada pengembangan model *misuse cases* untuk modul autopilot berdasarkan perlindungan kesalahan pengguna dan toleransi kesalahan seperti yang didefinisikan dalam ISO 25010 [11]. Model *misuse cases* yang dikembangkan kemudian dievaluasi dengan *walkthrough* dengan partisipan yang berbeda [12]. Masukan dari partisipan kemudian akan dibahas untuk menemukan *gap* antara *safety requirements* yang ditunjukkan dalam model yang dikembangkan dan harapan pengguna.

Untuk mendapatkan tujuan yang diharapkan, berikut rumusan masalah yang akan menjadi focus dari penelitian yang dilakukan:

- Seberapa baik *misuse cases*, baik secara diagram maupun tekstual, dalam menunjukkan potensi kegagalan dan kesalahan pengguna?
- Pengembangan seperti apa yang dapat dilakukan untuk memperbaiki metode *misuse cases*?

1.3 Tujuan

Untuk mencapai tujuan tersebut, berikut tujuan dari penelitian yang dilakukan ini:

- Mengetahui seberapa baik *misuse cases*, baik secara diagram maupun tekstual, dalam menunjukkan potensi kegagalan dan kesalahan pengguna.
- Menganalisis pengembangan seperti apa yang dapat dilakukan untuk memperbaiki metode *misuse cases*.

Penelitian yang dilakukan berkontribusi pada bidang rekayasa persyaratan perangkat lunak dalam bentuk pengetahuan tentang manfaat dari *misuse cases* untuk analisis *safety requirements*.

1.4 Organisasi Tulisan

Penulisan Tugas Akhir ini disusun sebagai berikut: bagian 2 menjelaskan karya dan teori yang berkaitan dengan penelitian ini. Bagian 3 menjelaskan aktivitas selama proses penelitian ini. Bagian 4 menyajikan hasil penelitian ini dan membahasnya. Akhirnya, bagian 5 menyimpulkan artikel ini dan mengusulkan arahan masa depan untuk penelitian dalam metode kasus penyalahgunaan.

