SafeDri: A mobile-based application for safety driving

H Maulid*, W Nurhidayat and S J Priyono
Department of Software Engineering, School of Applied Science, Telkom University, Jl. Telekomunikasi no 1, Terusan Buah Batu, Bandung 40257, Indonesia

*hariandimaulid@tass.telkomuniversity.ac.id

Abstract. Globally, around 1.3 million people die every year and more than 50 million people suffer permanent disability due to traffic accidents. This condition has worsened in many countries in the world in line with the high rate of ownership of motorized vehicles. The most common reasons for traffic accidents are distracted driving, over speeding, drunk driving and reckless driving. All those reasons are related with unsafe behaviour while driving. To cope with the problem, we developed a mobile based application that assists driver to drive more safely. The apps will collect information during a stipulated journey and recognize speeding facts, harsh acceleration, heavy breaking, and fuel consumption. While using this application, the drivers will learn how to drive safely, set goals and preferences, get alerts upon over limit violation, and receive feedback on their completed journeys. The results of a usability test on the application indicate that the app managed to identify misbehaviour while driving and rate driver behaviour based on journey history.

1. Introduction
Road safety is an issue that tends to emerge from year to year. Nowadays it has become a global problem and is not merely a matter of transportation but has become a social problem. According to World Health Organization (WHO) report, around 1.3 million people die every year globally and more than 50 million people suffer permanent disability due to traffic accidents [1,2]. The majority victims killed in traffic accidents are people between 15 and 44 years. This condition has worsened in many countries in the world in line with the high rate of ownership of motorized vehicles.

Some main contributors to traffic accident in South East Asian countries including unlicensed driver, traffic law violation, inadequate of comprehensive development of roads and other transport infrastructure, and rapid and unplanned urbanization. The most common reasons for traffic accidents are distracted driving, over speeding, drunk driving and reckless driving [3]. All those reasons are related with unsafe behaviour while driving. However, road traffic accidents are preventable. Some measures have been taken ranging from enforcing legislation on controlling speed and alcohol consumption, obligate the use of helmet and seatbelt, to designing safer roads and vehicles.

In the form of technological advancement, the development of in-vehicle sensing technology has contributed to measure driving style more objectively. Such system includes collision-avoidance, neglected and distracted driver monitoring system, automatic steering system, and blind spot warning [4]. However, despite the effectiveness of those systems in improving driver safety, the price was not quite affordable since they only found in a pricey vehicle. On the other hand, the advancement of mobile sensing platform on regular smartphone enables capturing and processing data from the real world in a cost-effective manner.
Some projects that incorporating mobile sensing for safety driving have been conducted such as SafeDrive [5] that automatically silent driver’s smartphone when car speed reach the maximum limit. It also rewards the driver for not texting while driving. The drawback of that application is that it only available in Romania. Another mobile project was DriveWell [6], a behaviour-based program from Cambridge Mobile Telematics, that accurately measure the driving performance and vehicle dynamics to make better driving. Like SafeDrive, DriveWell also will only well suited when it is used in the country where it originates due to different geographical situation and road infrastructure.

Furthermore, there are two main focuses on the application of Intelligent Transportation System (ITS), efficiency and safety application. The latter formerly used for avoiding or reducing traffic incidents as well as their impact. With ITS, vehicles can communicate with each other through Vehicle-to-Vehicle communication or Vehicle-to-Infrastructure technologies [7]. Vehicular Ad hoc NETworks (VANET) is the example of vehicle-to-vehicle ad hoc network [8]. One vehicle can inform other vehicles about some specific information such as traffic information, traffic accident, road conditions, and driving navigation [9-12]. Some concerns in VANET are cyber-security challenge [7] and trust management [13].

Some ITS safety application had incorporated Internet of Things (IoT) technology. Goshi et al developed an Assistant System for Safe Driving by Informative Supervision and Training (ASSIST) based on IoT architecture [14]. Their system comprised of an on-board system inside a vehicle (microprocessor, GPS, camera) to get various data including position, speed, and fuel consumption. These data will be sent to supervision system to calculate the summary of driving. Aital and Singh in [15] proposed an IoT-based risk assessment and warning system that assist the driver in maintaining a safe distance between vehicles and avoiding unsafe conditions on roads. IoT also had been used to build safety application for detecting vehicle obstacle [16], ensuring safe driving [17], and tracking hand and head movement while driving using wearable magnetics [18].

In mobile platform, there has been an active research on leveraging smartphone sensing to assist driver. SmartLDWS was the first LDWS on smartphone that warns the driver when a car moves out of the lane [19]. On the other hand, SignalGuru detects and predicts traffic signals with the camera and advises the driver to adjust speed for fuel consumption efficiency [20]. Mobile sensing also has been used in various projects to monitor driving style, pattern, and behaviour. DriveSafe [4], CarSafe [21], and Driving coach [22] are the three examples. DriveSafe detects inattentive driving behaviour such as drowsiness and distraction using computer vision and pattern recognition techniques. Similarly, CarSafe uses dual camera feature in smartphone to detect drowsiness, inattentive, tailgating, drifting, and ignoring blind spot. The front-facing camera monitor driver condition while the rear-facing camera tracking road conditions. Unlike the two, driving coach uses smartphone’s sensors and the state of the vehicle to detect driving pattern. It then will recommend new behaviours to the driver that supposedly to improve driving experience and fuel consumption efficiency. However, the most related work with our project on mobile safety driving application are SafeDrive [5] and DriveWell [6]. As aforementioned, the main drawback of those applications was they fit only specific geographical situation as well as road infrastructures where those applications originated from.

To address this issue, we argue for the possibility of utilizing a mobile based architecture on developing an application that assists the driver to drive more safely that well suited South-East Asian countries geographical and road infrastructures. The apps will collect information during a stipulated journey and recognize speeding facts, harsh acceleration, heavy-braking, and fuel consumption. While using this application, the drivers will learn how to drive safely, set goals and preferences, get alerts upon over limit violation, and receive feedback on their completed journeys. The goal of this work is to present the process of our mobile-based application development that targeted local users in order to gain user’s specific needs and expectations toward the application.

2. Project design and implementation

We used User-Centered Design method in designing user needs and expectations as well as project implementation. Figure 1 depicts all stages of project design and development. It started with collecting
data user needs and expectations, preliminary design, evaluation of design alternatives, prototypes, and final artifact.

![Diagram of design stages](image1)

**Figure 1.** User-centered design stages.

2.1. **Collecting data and preliminary design**

The first stage is collecting user needs and expectations using various data collection techniques including interview and questionnaires. We recruited 10 (ten) respondents to participate in this stage. We decided to combine this stage with preliminary design (stage two) for two reasons: to give initial picture design of the system to the respondents and to save the time. In doing so, we divided respondents into two groups with equal number. While the first group doing interview, other group filling the questionnaires. For the group who do an interview, we also provided a preliminary design that consists of the main features of the system as shown by figure 2.

![Prototype diagram](image2)

**Figure 2.** Preliminary design of the prototype.

The expected result of data collection is to determine the main goal of the system. All respondents agreed that the proposed system should manage to reduce traffic accident by minimizing its factors. Some actions identified as serious factors including over-speeding, drunk driving, distracted driving, and reckless driving. We also asked the respondents to comment on our proposed preliminary design whether it managed to achieve the goal or it required any improvement. At the end of these two stages we obtained user’s goal and new improvement on our preliminary design as follows:

- **Set your vehicle**
  - Car – type and year
  - Motorcycle – type and year

- **Start your journey**
  - Detect over-speeding
    - Include time and location
  - Detect heavy breaking
    - Include time and location

- **Stop your journey**
  - Draw the polyline for visual journey history on Gmap
  - Display Journey History
    - Display distance covered
    - Display number of violation (over speeding and heavy-braking)
2.2. System implementation
The good thing of user-centered design is to involve users throughout all processes from design phase to implementation phase including its evaluation in an iterative manner until fit users’ goal [23]. In this implementation phase, we did two iterations. The First iteration covers a sketch of low-fidelity and high-fidelity prototype. Prototype implementation and its evaluation conducted in the second iteration.

2.2.1. First iteration. In the first iteration, we discussed and refined low and high-fidelity prototype together with all respondents/users according to the result of preliminary design. We asked them to draw a sketch on the paper on expected interface. This low fidelity allowing respondents to create and modify interface straightforwardly. Upon agreed on a low-fidelity sketch, we designed its high-fidelity interface to be implemented at the next iteration. An example of low fidelity sketches along with its high-fidelity interface are shown in figure 3 and 4.

![Figure 3. A low fidelity sketch.](image)

![Figure 4. High-fidelity prototype.](image)

2.2.2. Second iteration. The second iteration consists of implementing low and high-fidelity interface into hard coding implementation.

![Figure 5. System architecture.](image)

Beforehand, we have determined to use Android mobile platform and Firebase as a Backend as a Service (BaaS) as implementation environment of this system. We used several sources of raw data namely smartphone sensors (GPS, accelerometer, Gyroscope, sound, light, and weather), OBD-II Dongle with Bluetooth (RPM, speed, pedal position, temperature, and fuel system), and other third party data or remote data (map, traffic info, location, weather, car workshop, and gas station). The detailed system architecture can be found in figure 5.
The Android application (SafeDri) will be the heart of the system. It will collect and pre-process raw data that derived from smartphone sensors, OBD-II, and remote data to obtain meaningful information (data acquisition). To produce meaningful data, the SafeDri should create data fusion modules which acquire and process raw data and turn them into meaningful information. For example, to obtain over speeding data, a simple module recalls raw speed, RPM, and location from OBD-II and internal sensor before then combines them with speed limit rule. Another example of data fusion module is when the app requires location-based service (nearest gas station or nearest car workshop) it can query location service like Google Map or OpenStreetMap based on GPS current location sensor.

Based on first iteration result, we turned high-fidelity prototype into system prototype named SafeDri. This app will collect information during stipulated journey such as over speeding and heavy braking. After each ended trip, the app will calculate the number of violations relating to over speed and heavy braking. Moreover, not only the total of violation displayed on the history, but also time and location, when and where the user made miss-driving-behaviour as displayed by figure 6 to figure 8.

![Figure 6. SafeDri main menu and features. Start the journey and display location.](image1)

![Figure 7. Recent journey history and violation.](image2)

![Figure 8. Detail of violation data (heavy braking).](image3)

3. Discussion
While developing heavy breaking detection, on the other hand, we encountered long debatable discussion. As heavy breaking could be categorized both for normal condition to avoid crash or abnormal condition (violation), however, any form of immediate breaking (extreme lower speeding until stopping) could lead to the series of collision. Imagine when a car speed reaches 60 km/hour and suddenly stops on avoiding a pedestrian, there is the possibility of other vehicles behind that car to hit and leading to the series of unintended accident.

To validate the system, we went through usability testing with System Usability Scale (SUS) to measure the usability aspect of the app [24]. SUS consists of 10 (ten) questions that give global subjective assessment view toward the UI/UX of the app. This test conducted during the second iteration phase. After the system being implemented, we asked 10 (ten) respondents to use and explore SafeDri app in one-week period and record the states.
Table 1. Result of SUS score.

| Respondents | USERS Answer per-question | SUS Average per respondents | Final SUS Score |
|-------------|----------------------------|-----------------------------|-----------------|
| User 1      | 5, 1, 5, 2, 4, 2, 5, 2, 5, 2 | 87.5                        |                 |
| User 2      | 5, 2, 4, 1, 5, 1, 5, 2, 4, 3 | 85.5                        |                 |
| User 3      | 5, 2, 5, 2, 5, 2, 5, 1, 5, 1 | 92.5                        |                 |
| User 4      | 5, 1, 4, 2, 5, 1, 4, 1, 5, 2 | 90.0                        |                 |
| User 5      | 4, 2, 5, 1, 5, 2, 4, 1, 5, 2 | 87.5                        |                 |
| User 6      | 5, 2, 5, 2, 5, 2, 5, 2, 4, 1 | 87.5                        |                 |
| User 7      | 4, 3, 4, 1, 5, 1, 5, 1, 4, 2 | 85.0                        |                 |
| User 8      | 5, 1, 5, 2, 5, 2, 5, 2, 5, 1 | 92.5                        |                 |
| User 9      | 5, 1, 3, 1, 4, 2, 5, 2, 5, 1 | 87.5                        |                 |
| User 10     | 5, 1, 4, 1, 5, 2, 5, 2, 4, 1 | 90.0                        |                 |

Next, we asked them to fill out SUS test based on user experience towards application in previous week. We use 5-likert scale with 5 as strongly agree and down to 1 as strongly disagree as the answer score for every question. The calculation method is described as follows:

- For odd numbered questions (1, 3, 5, 7 and 9), subtract 1 from the score.
- For even numbered questions (2, 4, 6, 8 and 10), subtracts their value from 5.
- SUS Average per-respondent by adding up all total score then multiply this by 2.5
- Final SUS score calculated by totalling all SUS average per-respondent and then divided by the number of respondents.

![Figure 9. Acceptability of SAFEDRI app.](image)

Table 1 shows the final average SUS score as users’ feedback. It reached 88.5 beyond the minimal target which is 68. The test score expressed excellent acceptable range (B) as depicted by figure 9. However, since this project is just the first part (4 months) out of a three-year ambitious project, there are some possibilities to add some features and technologies advocated in the app. Some possible features including mood and fatigue detection, lane trajectory detection, collision avoidance, distracted driver detection, traffic congestion detection, motorway and hilly road detection, as well as city and countryside location determination. Consequently, some sensors and technologies should be considered such as Internet of Things (IoT), dual camera facility, not to mention machine learning and data science.

4. Conclusion
SafeDri takes advantage of the development of mobile sensing that provides an identification of misbehavior while driving and rate driver behaviour based on journey history especially when the driver tries to break the limit of speed or waste the fuel consumption and do a sudden stop. Although the SUS test score reached excellent, it still needs some improvement, especially in terms of features on various
condition detection along with their technological advancement. We plan to implement some features that significantly affect the driver style and behaviour such as mood and fatigue, drowsiness, distraction, and surrounding environment detection.

References
[1] Peden M, Scurfield R, Sleet D, Mohan D, Hyder AA, Jarawan E, and Mathers C 2004 World report on road traffic injury prevention (Geneva: World Health Organization).
[2] Hu W, Hu X, Deng J, Zhu C, Fotopoulos G, Ngai E C H and Leung V C M 2014 Proc. Int. on Middleware for Context-Aware Applications in the IoT (Bordeaux) I pp 19 – 24.
[3] King E L 2017 available online: https://www.huffpost.com/entry/top-15-causes-of-car-accidents_b_11722196 last accessed 20 July 2019.
[4] Bergasa L M, Almería D, Almazán J, Yebes JJ and Arroyo R 2014 Drivesafe: An app for alerting inattentive drivers and scoring driving behaviors: Proc. Int. 2014 IEEE Intelligent Vehicles symposium proceedings (Miami) pp 240-245.
[5] Cobalas T 2014 SafeDrive app rewards your good behavior while driving available online: https://devpost.com/software/safedrive-app last accessed 20 July 2019.
[6] Cambridge Mobile Telematics DriveWell: A behavior-based program available online: https://www.cmtelematics.com/drivewell/ last accessed 20 July 2019.
[7] Mukisa S S and Rashid A 2014 Cyber-security challenges of agent technology in intelligent transportation systems. In Proceedings of the 1st International Workshop on Agents and CyberSecurity (Paris) pp 9.
[8] Intelligent Transportation Systems Committee 2011 IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE)-Multichannel Operation. IEEE, Standard, pp.1609-4.
[9] Choffnes D R and Bustamante F E 2005 An integrated mobility and traffic model for vehicular wireless networks. In Proc. Int. of the 2nd ACM international workshop on Vehicular ad hoc networks (Cologne) (New York: ACM) pp. 69-78.
[10] Härri J, Filiali F, Bonnet C and Fiore M 2006 VanetMobiSim: generating realistic mobility patterns for VANETs. In Proc. Int. of the 3rd international workshop on Vehicular ad hoc networks (Los Angeles) (New York: ACM) pp. 96-97.
[11] Lochert C, Scheuermann B, Wewetzer C, Luebke A and Mauve M 2008 Data aggregation and roadside unit placement for a vanet traffic information system. In Proc Int. of the fifth ACM international workshop on Vehicular Inter-NETworking (San Francisco) (New York: ACM) pp. 58-65.
[12] Cops M 2006 Vehicle infrastructure integration (VII)-bringing vehicle adhoc network technology on the road. In the Third ACM International Workshop on Vehicular Ad Hoc Networks (New York: ACM).
[13] Ma S, Wolfson O and Lin J 2011 A survey on trust management for intelligent transportation System. In Pro. Int. of the 4th ACM SIGSPATIAL International Workshop on Computational Transportation Science (Chicago) (New York: ACM) pp. 18-23.
[14] Goshi K, Hayahi M, Sumida Y and Matsunaga K 2016 An Iot-based safe driving management System. In Proc. Int. on Mobile and Ubiquitous Systems: Computing, Networking andservices (Hiroshima) vol 13 (New York: ACM) pp 280-281.
[15] Aithal A and Singh S 2018 IoT Based Driving Risk Assessment and Warning System. In 2018 International Conference on Smart Systems and Inventive Technology (ICSSIT) (Tirunelveli, India) IEEE pp. 282-286.
[16] Umakirthika D, Pushparani P and Rajkumar M V 2018 Internet of Things in Vehicle Safety—Obstacle Detection and Alert System. Int. J. Eng. Comput. Sci. 7 pp.23540-23551.
[17] Al Mamun M A, Puspo J A and Das A K. 2017 An intelligent smartphone-based approach using IoT for ensuring safe driving. In 2017 International Conference on Electrical Engineering and Computer Science (ICECOS)(Palembang) IEEE pp. 217-223.
[18] Huang H, Chen H and Lin S 2019 MagTrack: Enabling Safe Driving Monitoring with Wearable Magnetics. In Proc. of the 17th Annual International Conference on Mobile Systems, Applications, and Services (Seoul) (New York: ACM) pp. 326-339.

[19] Lan M, Rofouei M, Soatto S and Sarrafzadeh M 2009 SmartLDWS: A robust and scalable lane departure warning system for the smartphones. In 2009 12th International IEEE Conference on Intelligent Transportation Systems (St. Louis) IEEE pp. 1-6.

[20] Koukoumidis E, Peh L S and Martonosi M R 2011 SignalGuru: leveraging mobile phones for collaborative traffic signal schedule advisory. In Proc of the 9th international conference on Mobile systems, applications, and services (Bathesda) (New York: ACM) pp. 127-140.

[21] You C W, Lane N D, Chen F, Wang R, Chen Z, Bao T J, Montes-de-Oca M, Cheng Y, Lin M, Torresani L and Campbell AT 2013 Carsafe app: Alerting drowsy and distracted drivers using dual cameras on smartphones. In Proc. of the 11th annual international conference on Mobile systems, applications, and services (Taipei) (New York: ACM) pp. 13-26.

[22] Araújo R, Igreja A, De Castro R and Araujo R E 2012 Driving coach: A smartphone application to evaluate driving efficient patterns. In 2012 IEEE Intelligent Vehicles Symposium (Alcala de Henares) IEEE pp. 1005-1010.

[23] Maulid H, Azimi I and Fauzi A H 2018 A user-centered design based collaborative system for Jum’a Preacher Scheduling. Int. Journal on Islamic Applications in Computer Science and Technology (IJASAT) UK.

[24] Borsci S, Federici S and Lauriola M 2009 On the dimensionality of the System Usability Scale: a test of alternative measurement models. Cognitive processing vol 10 issue 3 (Springer: Berlin) pp.193-197.