**System Test Plan**

**For**

**EcoCAR**

Team member: Teja Tiriveedhi, Nathan Rose, Sabrina Yepez, David Fadini, Raymond Kwasneski

|  |  |
| --- | --- |
| Version/Author | Date |
| 1/Teja Tiriveedhi | 11/5/2020 |
|  |  |
|  |  |
|  |  |

Table of Contents

[1. Introduction 2](#_Toc530500519)

[1.1 Purpose 2](#_Toc530500520)

[1.2 Objectives 2](#_Toc530500521)

[2. Functional Scope 2](#_Toc530500522)

[3. Overall Strategy and Approach 2](#_Toc530500523)

[3.1 Testing Strategy 2](#_Toc530500524)

[3.2 System Testing Entrance Criteria 2](#_Toc530500525)

[3.3 Testing Types 2](#_Toc530500526)

[3.4 Suspension Criteria and Resumption Requirements 3](#_Toc530500527)

[4. Execution Plan 3](#_Toc530500528)

[4.1 Execution Plan 3](#_Toc530500529)

[5. Traceability Matrix & Defect Tracking 3](#_Toc530500530)

[5.1 Traceability Matrix 3](#_Toc530500531)

[5.2 Defect Severity Definitions 3](#_Toc530500532)

[6. Environment 4](#_Toc530500533)

[6.1 Environment 4](#_Toc530500534)

[7. Assumptions 4](#_Toc530500535)

[8. Risks and Contingencies 4](#_Toc530500536)

[9. Appendices 4](#_Toc530500537)

# Introduction

## Purpose

## This document is a test plan for the EcoCAR User Interface (UI) System Testing, produced by the EcoCAR UI Senior Design Team. It describes the testing strategy and approach to the team will use to verify the application meets the established requirements of the project prior to release.

## Objectives

* The system meets the requirements, specifications and the project rules.
* The system supports the intended project functions and achieves the required standards.
* The system satisfies the Entrance Criteria for User Acceptance Testing.

# Functional Scope

The functional scope for testing the EcoCAR UI System is solely focused on the user interface display and information presented. The system will display the relative location of all nearby cars to the driver.

# Overall Strategy and Approach

## Testing Strategy

The EcoCAR UI System Testing will include testing of all functionalities that are in the scope (Refer Functional Scope Section). System testing activities will include the testing of new functionalities, modified functionalities, screen level validations, workflows, functionality access, and testing of internal & external interfaces.

## System Testing Entrance Criteria

* The system is receiving power
* The system is properly connected.

## Testing Types

### Usability Testing

User interface attributes, cosmetic presentation and content will be tested for accuracy and general usability. The goal of Usability Testing is to ensure that the User Interface is easy to use and provides the driver with consistent access and navigation through the functions of the system.

### Functional Testing

The objective of the tests is to ensure that each element of the system meets the functional requirements of the project as outlined in the:

* Project / Functional Requirements.
* Project rules or conditions.
* Other functional documents produced during the project i.e. resolution to issues/change requests/feedback.

## Suspension Criteria and Resumption Requirements

### Suspension Criteria

Testing will be suspended if the incidents found will not allow further testing of the system under test. If testing is halted, and changes are made to the hardware, software or database, it is up to the Testing Manager to decide whether the test plan or part of the test plan will be re-executed.

### Resumption Requirements

Resumption of testing will be possible when the functionality that caused the suspension of testing has been retested successfully.

# Execution Plan

## Execution Plan

The software will be tested on a raspberry pi, the tests will be run through and if a failure occurs, the issue must be fixed and the test rerun.

# Traceability Matrix & Defect Tracking

## Traceability Metrix

List of requirements (corresponding test cases)

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement | Test Case | Pass/Fail | Comments |
| The system shall turn on when car ignition is on. | The system powers on when the car turns on |  | We are not going to be able to test this requirement this semester, as we are working on ensuring the system software will function. We are not focusing on hardware. |
| The system shall turn off when the car ignition is off. | The system powers off when the car turns off. |  | We are not going to be able to test this requirement this semester, as we are working on ensuring the system software will function. We are not focusing on hardware. |
| The system shall be updating information to the display every 100 milliseconds. | The call to update the UI screen is run at least once every 100 milliseconds. |  |  |
| The system shall display the relative position of nearby obstacles to the driver. | Give replicated data that represents the relative position of nearby obstacles. |  |  |
| The system shall display the lane type. | Different types of lanes such as solid white line, dashed line, solid yellow lines will be replicated to the UI. |  |  |
| The system shall display lane location. | Give replicated data that represents the location of the lanes. |  |  |
| The system shall display cruise control speed. | Give replicated data that represents the speed from the ROS. |  |  |
| The system shall provide a brief training video on how the ACC works to the driver. | Training video will be displayed on the UI and played from start to end time successfully. |  |  |
| The system shall get the driver’s attention by turning the top of the screen red when the EcoCAR detects a vehicle in front of it less than (V\* R) feet away.  Where:  V = speed of car (mi/hr)  R = ratio (2hr) | Red color is displayed instantaneously when the EcoCAR detects a vehicle in front of it less than (V \* R) feet away. |  |  |
| The system shall display any issues that the on-board computer detects in the ACC system. | Display replicated defects such as the ACC is not functioning alert. |  |  |
| The system shall provide drivers a way to re-watch the training video. | The system will allow drivers to re-watch the training video by allowing the driver to press a button to play the video. |  |  |
| The system will allow drivers to switch from a light mode to a dark mode. | The system will adapt to the correct lighting environment by the amount of light around the car. |  |  |

## Defect Severity Definitions

|  |  |
| --- | --- |
| **Critical** | The defect causes a catastrophic or severe error that results in major problems that makes the system unavailable to the driver. High effort is required to fix the defect. Examples of a critical defect are as follows:   * The system abends. * Data cannot flow through a project function/lifecycle. * Data is corrupted or cannot post to the database. |
| **Medium** | A defect that does not seriously impair system function can be categorized as a medium Defect. Moderate effort is required to fix the defect. Examples of a medium defect are as follows:   * Navigation is incorrect. * Field labels are not consistent with global terminology. |
| **Low** | The defect is cosmetic or has little to no impact on system functionality. Little effort is required to fix the defect. Examples of a low defect are as follows:   * Repositioning of fields on screens. * The font on reports is incorrect. |

# Environment

## Environment

* A Raspberry Pi will be used for testing the UI. The UI dashboard interface will be implemented on the EcoCAR at a later stage. A prototype is currently being tested independently in a Linux operating system. Later, it will be implemented using Raspberry PI on the EcoCAR. All the physical environment requirements and conditions will be tested at a later stage once the software is integrated with the Hardware.

# Assumptions

* One of the assumptions is that the test plans will be tested independently on a platform by the designer, it will not be tested on the hardware. The EcoCAR has yet to be interfaced with the UI.
* Another assumption is that the driver does not know how the UI nor the EcoCAR works in terms of functionally. All the requirements will be tested with the assumption that the driver is using it for the first time and is not aware of the UI and its functional features. These requirements will be tested by assuming that drivers do not own the EcoCAR.

# Risks and Contingencies

One risk is that the interface between the driver and the EcoCAR will change because there is a separate team working on generating data for the UI team. This could lead to a conflict in data transmission. This has been accounted for by making the system modular and easy to change based on the data that the EcoCAR is giving us.

# Appendices

# 