TEN | Traffic Engine

Plano de Testes

**Versão 1.00**

06/07/2009

# Responsáveis

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Antônio Cláudio Goméz de Sousa

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Fernando Seabra Chirigati

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Rafael Shinji Aoki Kikuchi

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Talita Lopes Gomes

# Relatório de Mudanças

TEN – Traffic Engine

**Versão 1.00 – 06/07/2009**

Criação do Documento

# Índice

[1. Introdução 1](#_Toc234565355)

[1.1 Finalidade 1](#_Toc234565356)

[1.2 Definições, Acrônimos e Abreviaturas 1](#_Toc234565357)

[1.3 Referências 1](#_Toc234565358)

[2. Descrição Geral 2](#_Toc234565359)

[2.1 Itens 2](#_Toc234565360)

[2.1.1 Simulador 2](#_Toc234565361)

[2.1.2 Interface 2](#_Toc234565362)

[2.1.3 Atualizador 2](#_Toc234565363)

[2.1.4 Dados do Mapa e Dados dos Veículos 2](#_Toc234565364)

[2.2 Requisitos 2](#_Toc234565365)

[2.3 Visão geral 2](#_Toc234565366)

[2.4 Suspensão ou conclusão 3](#_Toc234565367)

[2.5 Ambiente 3](#_Toc234565368)

[2.6 Tarefas e cronograma 3](#_Toc234565369)

[2.7 Riscos e gerenciamento 3](#_Toc234565370)

[3. Especificações dos testes 4](#_Toc234565371)

[3.1 Confecção do Mapa 4](#_Toc234565372)

[3.1.1 Identificador 4](#_Toc234565373)

[3.1.2 Características 4](#_Toc234565374)

[3.1.3 Refinamento 4](#_Toc234565375)

[3.1.4 Identificador de caso de teste 4](#_Toc234565376)

[3.2 Definição de Parâmetros 4](#_Toc234565377)

[3.2.1 Identificador 5](#_Toc234565378)

[3.2.2 Características 5](#_Toc234565379)

[3.2.3 Refinamento 5](#_Toc234565380)

[3.2.4 Identificador de caso de teste 5](#_Toc234565381)

[3.3 Simulação 5](#_Toc234565382)

[3.3.1 Identificador 5](#_Toc234565383)

[3.3.2 Características 5](#_Toc234565384)

[3.3.3 Refinamento 5](#_Toc234565385)

[3.3.4 Identificador de caso de teste 6](#_Toc234565386)

[4. Casos de teste 7](#_Toc234565387)

[4.1 Criação de uma rua com um número válido de vias 7](#_Toc234565388)

[4.1.1 Identificador 7](#_Toc234565389)

[4.1.2 Itens 7](#_Toc234565390)

[4.1.3 Entradas e Saídas 7](#_Toc234565391)

[4.2 Criação de uma rua com um número inválido de vias 7](#_Toc234565392)

[4.2.1 Identificador 7](#_Toc234565393)

[4.2.2 Itens 7](#_Toc234565394)

[4.2.3 Entradas e Saídas 7](#_Toc234565395)

[4.3 Criação de uma rua conectada a outra 7](#_Toc234565396)

[4.3.1 Identificador 7](#_Toc234565397)

[4.3.2 Itens 7](#_Toc234565398)

[4.3.3 Entradas e Saídas 7](#_Toc234565399)

[4.4 Remoção de uma rua 8](#_Toc234565400)

[4.4.1 Identificador 8](#_Toc234565401)

[4.4.2 Itens 8](#_Toc234565402)

[4.4.3 Entradas e Saídas 8](#_Toc234565403)

[4.5 Criação de um semáforo em uma rua com uma temporização válida 8](#_Toc234565404)

[4.5.1 Identificador 8](#_Toc234565405)

[4.5.2 Itens 8](#_Toc234565406)

[4.5.3 Entradas e Saídas 8](#_Toc234565407)

[4.6 Criação de um semáforo em uma rua com uma temporização inválida 8](#_Toc234565408)

[4.6.1 Identificador 8](#_Toc234565409)

[4.6.2 Itens 8](#_Toc234565410)

[4.6.3 Entradas e Saídas 8](#_Toc234565411)

[4.7 Remoção de um semáforo em uma rua 9](#_Toc234565412)

[4.7.1 Identificador 9](#_Toc234565413)

[4.7.2 Itens 9](#_Toc234565414)

[4.7.3 Entradas e Saídas 9](#_Toc234565415)

[4.8 Criação de um novo mapa 9](#_Toc234565416)

[4.8.1 Identificador 9](#_Toc234565417)

[4.8.2 Itens 9](#_Toc234565418)

[4.8.3 Entradas e Saídas 9](#_Toc234565419)

[4.9 Definição de uma nova velocidade máxima, válida, de uma rua 9](#_Toc234565420)

[4.9.1 Identificador 9](#_Toc234565421)

[4.9.2 Itens 9](#_Toc234565422)

[4.9.3 Entradas e Saídas 9](#_Toc234565423)

[4.10 Definição de uma nova velocidade máxima, inválida, de uma rua 9](#_Toc234565424)

[4.10.1 Identificador 10](#_Toc234565425)

[4.10.2 Itens 10](#_Toc234565426)

[4.10.3 Entradas e Saídas 10](#_Toc234565427)

[4.11 Definição de parâmetros gerais válidos 10](#_Toc234565428)

[4.11.1 Identificador 10](#_Toc234565429)

[4.11.2 Itens 10](#_Toc234565430)

[4.11.3 Entradas e Saídas 10](#_Toc234565431)

[5. Procedimentos de teste 11](#_Toc234565432)

[5.1 Procedimento 1 11](#_Toc234565433)

[5.1.1 Identificador 11](#_Toc234565434)

[5.1.2 Finalidade 11](#_Toc234565435)

[5.1.3 Necessidades especiais 11](#_Toc234565436)

[5.1.4 Ações 11](#_Toc234565437)

[5.1.5 Relatórios 11](#_Toc234565438)

# Introdução

## Finalidade

O TEN – Traffic Engine – é um software que simula o tráfego de veículos em uma região determinada pelo usuário. Dessa forma, análises podem ser realizadas para que congestionamentos freqüentes de uma grande cidade, por exemplo, possam ser amenizados. A região determinada pelo usuário deve ser confeccionada pelo mesmo através do próprio aplicativo.

A finalidade do plano de testes diz respeito a verificar a existência de erros, falhas e inconsistências no software, para que estes sejam evitados, garantindo a qualidade do sistema desenvolvido. Ele está dirigido para a equipe de desenvolvimento do aplicativo e para o gerente de qualidade do projeto, o professor Antônio Cláudio Goméz de Sousa.

## Definições, Acrônimos e Abreviaturas

* *Nó*: Ponto que define uma posição na área delimitada para a confecção de um mapa. Ele corresponde a uma extremidade de uma rua.
* *Aresta*: Conexão entre dois nós que representa uma rua do mapa.

## Referências

* *Plano de Gerenciamento de Projeto de Software – PGPS*

Versão: 1.00

Data: 01/04/2009

Responsável: Equipe de desenvolvimento do projeto

* *Especificação de Requisitos de Software – ERS*

Versão: 1.10

Data: 17/06/2009

Responsável: Equipe de desenvolvimento do projeto

* *Manual do Usuário*

Versão: 2.01

Data: 06/07/2009

Responsável: Equipe de desenvolvimento do projeto

* *Projeto de Software*

Versão: 1.10

Data: 06/07/2009

Responsável: Equipe de desenvolvimento do projeto

* *Implementando um Simulador de Tráfego Urbano para uma Interseção com Semáforos*

Glleddson Fryttys Menezes Leite, Antônio César Baleeiro Alves

Departamento de Computação – Universidade Católica de Goiás – UCG

Site da referência: *http://wsmartins.net/ermacs/trabalho\_18.pdf*

# Descrição Geral

## Itens

O sistema do TEN foi dividido em cinco módulos, os quais foram descritos no Projeto de Software do aplicativo. Assim, os testes a serem realizados têm o objetivo de identificar erros e falhas em cada um desses módulos, que estão descritos abaixo, juntamente com seus critérios de falha e correção.

### Simulador

Este módulo é responsável pelos algoritmos (cujo modelo encontra-se referenciado na Seção 1.3) e por todas as estruturas necessárias à simulação. Dessa forma, é imprescindível que a simulação ocorra de acordo com os algoritmos desenvolvidos. Caso alguma inconsistência seja encontrada, como, por exemplo, dois automóveis não respeitarem a distância de segurança determinada, a implementação dos algoritmos deve ser reavaliada.

### Interface

Este módulo é responsável por apresentar a interface gráfica para o usuário, por onde haverá todas as interações com o software. Assim, a interface deve estar consistente com os comandos acessados pelo usuário. Caso o usuário clique no botão de adicionar uma nova rua, que se encontra especificado no Manual do Usuário, esta ferramenta deve ser apresentada; caso isto não aconteça, o módulo apresenta uma falha em sua codificação.

### Atualizador

Este módulo é responsável por enviar pedidos de atualização periódicos à interface enquanto uma simulação está em andamento. Com isso, pretende-se manter uma sensação de continuidade para o usuário. Caso uma simulação se apresente ao usuário de forma não-contínua, ou seja, com intervalos de tempo significativos entre dois redesenhos, há indícios de que houve alguma falha no módulo.

### Dados do Mapa e Dados dos Veículos

Estes dois módulos são responsáveis por manter as estruturas de dados que contêm as informações referentes ao mapa e aos veículos da simulação, respectivamente. Dessa forma, assim como os parâmetros determinados pelo usuário, as informações obtidas durante a simulação não podem ser perdidos enquanto a seção estiver aberta.

## Requisitos

Os testes a serem realizados dizem respeito em verificar se os casos de uso do sistema, apresentados na Especificação de Requisitos de Software (ERS), funcionam sem falhas. Dessa forma, pretende-se garantir que todos os requisitos estabelecidos anteriormente sejam atendidos pelo software sem a presença de erros.

Devido à grande quantidade de casos de uso existentes, optou-se por não se considerar alguns deles; particularmente, não foram selecionados os Casos de Uso 3 (Aumentar Zoom) e 4 (Diminuir Zoom), uma vez que eles não apresentam alterações crucias nos resultados do software.

## Visão geral

As atividades de testes serão realizadas pelos integrantes da equipe que não estiveram demasiadamente ligados à codificação do sistema. Não será usada nenhuma ferramenta específica, uma vez que não há necessidade.

Para que alguns testes sejam realizados, arquivos *log* serão criados, de modo que informações importantes sejam coletadas internamente ao sistema.

## Suspensão ou conclusão

Haverá suspensão dos testes caso alguma falha seja encontrada. Neste caso, um dos integrantes da equipe de desenvolvimento deve corrigi-la para que os testes sejam retomados. Depois de realizados todos os testes e corrigidos todos os erros que o software eventualmente apresentou, pode-se concluir a etapa dos testes.

## Ambiente

Os testes serão realizados no mesmo ambiente em que o sistema estiver instalado, como, por exemplo, nos computadores pessoais da equipe de desenvolvimento do software.

## Tarefas e cronograma

Segundo o Plano de Gerenciamento de Projeto de Software (PGPS), a etapa de testes é composta por cinco dias. Dessa forma, foram identificadas três tarefas principais, cada uma delas relacionada a determinados requisitos. A tabela a seguir mostra as tarefas identificadas, juntamente com seus requisitos associados e seu prazo de duração estipulado.

|  |  |  |
| --- | --- | --- |
| Nome | Requisitos Associados | Prazo de Duração |
| Confecção do Mapa | Casos de Uso 1, 2, 7, 8 e 10 | 2 dias |
| Definição de Parâmetros | Casos de Uso 5, 6 e 9 | 1 dia |
| Simulação | Casos de Uso 11, 12, 13, 14, 15 e 16 | 2 dias |

Os prazos de duração envolvem não só a realização dos testes, como também a identificação e solução dos erros encontrados, caso haja.

## Riscos e gerenciamento

O principal risco existente diz respeito à presença de um erro que não foi identificado através dos testes realizados. Dessa forma, os usuários do sistema que encontrarem este erro devem reportá-lo à equipe de desenvolvimento do software para que ele seja corrigido imediatamente. Caso seja um erro que comprometa o uso e a eficiência de muitas funções do software, a correção do mesmo será enviada aos usuários o mais rápido possível; caso contrário, essa correção estará presente em uma próxima versão do aplicativo.

Além disso, há o risco do prazo estipulado para os testes não ser suficiente para a realização dos mesmos. Esse risco foi apresentado considerando o projeto como um todo no Plano de Gerenciamento de Projeto de Software (PGPS), na Seção 5.4. O plano de Risk Mitigation Monitoring and Management (RMMM) **R2** apresenta as diretrizes que devem ser tomadas caso o prazo se mostre insuficiente.

# Especificações dos testes

## Confecção do Mapa

### Identificador

O identificador único para o teste Confecção do Mapa é **T1**.

### Características

Neste teste, serão avaliados os seguintes casos de uso, os quais estão associados à confecção da região sobre a qual a simulação irá rodar:

* Caso de Uso 1 (Criar Rua)
* Caso de Uso 2 (Remover Rua)
* Caso de Uso 7 (Semaforizar um Ponto)
* Caso de Uso 8 (Remover Semáforo)
* Caso de Uso 10 (Criar Novo Mapa)

### Refinamento

Para que o teste **T1** tenha sucesso, alguns requisitos principais devem ser atendidos nos seguintes módulos do sistema:

* Interface: este módulo deve apresentar as ferramentas da interface correspondentes aos comandos acessados pelo usuário; à medida que o mapa é confeccionado, ele deve ser apresentado sem erros relacionados ao seu desenho, como, por exemplo, bordas das ruas e semáforos bem delimitados e intersecções das ruas bem representadas; por fim, quando um novo mapa for criado, a área de desenho e simulação deve ser limpa para que um novo mapa possa ser confeccionado.
* Dados do Mapa: quando os Casos de Uso 1 e 7 são executados, janelas de configuração são apresentadas ao usuário, caso ele queira modificar alguns parâmetros; se esses parâmetros são modificados, e os mesmos são válidos, eles devem ser imediatamente alterados internamente no sistema pelo módulo Dados do Mapa; um erro é identificado se essa alteração dos parâmetros não é capturada pelo sistema.

### Identificador de caso de teste

Os casos de teste associados a esta especificação são:

* C01 – Criação de uma rua com um número válido de vias
* C02 – Criação de uma rua com um número inválido de vias
* C03 – Criação de uma rua conectada a outra
* C04 – Remoção de uma rua
* C05 – Criação de um semáforo em uma rua com uma temporização válida
* C06 – Criação de um semáforo em uma rua com uma temporização inválida
* C07 – Remoção de um semáforo em uma rua
* C08 – Criação de um novo mapa

## Definição de Parâmetros

### Identificador

O identificador único para o teste Definição de Parâmetros é **T2**.

### Características

Neste teste, estão associados os casos de uso relacionados à definição de parâmetros gerais da simulação e à alteração de outros parâmetros específicos. Os casos de uso estão listados abaixo.

* Caso de Uso 5 (Alterar Velocidade Máxima da Rua)
* Caso de Uso 6 (Definir Parâmetros Gerais)
* Caso de Uso 9 (Alterar Temporização do Semáforo)

### Refinamento

O sucesso do teste **T2** depende do atendimento a alguns requisitos pelo módulo denominado *Dados do Mapa*, como exposto abaixo.

* Dados do Mapa: este módulo deve capturar a alteração dos parâmetros para que eles sejam modificados internamente no sistema; além disso, caso esses parâmetros sejam inválidos, sua alteração não seve ser permitida.

### Identificador de caso de teste

Os casos de teste associados ao teste **T2** encontram-se listados abaixo.

* C12 – Definição de uma nova velocidade máxima, válida, de uma rua
* C13 – Definição de uma nova velocidade máxima, inválida, de uma rua
* C14 – Definição de parâmetros gerais válidos
* C15 – Definição de parâmetros gerais, em que um deles seja inválido
* C16 – Definição de parâmetros gerais inválidos
* C17 – Definição de uma nova temporização, válida, de um semáforo
* C18 – Definição de uma nova temporização, inválida, de um semáforo

## Simulação

### Identificador

O identificador único para o teste Simulação é **T3**.

### Características

Neste teste, estão associados os casos de uso que estão relacionados com a simulação do tráfego de automóveis na região desenhada pelo usuário. Eles encontram-se listados abaixo.

* Caso de Uso 11 (Iniciar Simulação)
* Caso de Uso 12 (Pausar Simulação)
* Caso de Uso 13 (Retomar Simulação)
* Caso de Uso 14 (Parar Simulação)
* Caso de Uso 15 (Reiniciar Simulação)
* Caso de Uso 16 (Gerar Relatório)

### Refinamento

Alguns pontos devem ser verificados, em relação aos módulos do sistema, para que o teste **T3** tenha sucesso. Esses pontos estão listados abaixo.

* Simulador: este módulo deve garantir que a simulação em andamento esteja consistente com os algoritmos desenvolvidos; ou seja, os carros devem seguir as ruas, um carro não pode bater em outro, caso haja um semáforo, este deve ser respeitado, e assim por diante; além disso, ele deve garantir quando um comando de simulação está apto a ser executado (por exemplo, uma simulação não pode ser retomada se ela não está pausada).
* Interface: este módulo deve estar sempre se atualizando, através de pedidos do módulo Atualizador, enquanto uma simulação está em curso, para que a mesma se apresente de forma correta ao usuário, sem falhas; ou seja, não deve haver erros no desenho do mapa e dos veículos durante o curso de uma simulação.
* Atualizador: para que o módulo Interface apresente a simulação com uma sensação de cadência ao usuário, é necessário que o Atualizador esteja em constante atividade, já que ele indica à Interface a necessidade de atualizar os dados da simulação; assim, caso haja erros nesse módulo, a visualização da simulação pode ficar demasiadamente prejudicada.
* Dados do Mapa: a simulação em curso usa os parâmetros definidos pelos usuários, como a distância de segurança, por exemplo; logo, este módulo deve garantir a integridade desses parâmetros, de modo que o Simulador não obtenha dados inconsistentes para a simulação; além disso, grande parte desses dados é usada para a geração do relatório.
* Dados dos Veículos: os dados referentes aos veículos, como velocidade, não são determinados pelos usuários; porém, eles são indiretamente determinados por outros parâmetros, como distância de segurança, presença de semáforos, entre outros; logo, o módulo Dados dos Veículos não pode possuir erros no acesso a esses parâmetros; os dados dos veículos também são usados para a geração do relatório, que não pode estar inconsistente.

### Identificador de caso de teste

Os casos de teste associados a esta especificação são:

* C19 – Início de uma simulação
* C20 – Retomada de uma simulação em andamento
* C21 – Pausa de uma simulação
* C22 – Retomada de uma simulação pausada
* C23 – Término de uma simulação em andamento
* C24 – Término de uma simulação pausada
* C25 – Reinício de uma simulação em andamento
* C26 – Reinício de uma simulação pausada
* C27 – Geração de um relatório durante uma simulação em andamento
* C28 – Geração de um relatório durante uma simulação pausada
* C29 – Geração de um relatório sem a existência de uma simulação anterior

# Casos de teste

Todos os casos abaixo especificados supõem um ambiente de teste em que o TEN esteja instalado corretamente. Não há necessidades especiais em relação a hardware para esse ambiente, mas é necessário que o sistema operacional seja o Microsoft Windows, e que o framework .NET esteja instalado e atualizado.

## Criação de uma rua com um número válido de vias

### Identificador

O identificador único do caso de teste “Criação de uma rua com um número válido de ruas” é **C01**.

### Itens

Este caso se propõe a testar o módulo Interface, a fim de verificar se a rua é corretamente desenhada, e o módulo Dados do Mapa, para verificar se a informação do número de vias é devidamente obtida pelo sistema.

### Entradas e Saídas

A entrada para esse caso de teste é o uso da ferramenta que cria uma rua no mapa e um número de vias que seja do tipo *inteiro*.

Como saída, o sistema deve aceitar o valor do número de vias, sem que uma mensagem de erro seja produzida, além de desenhar corretamente a rua no mapa.

## Criação de uma rua com um número inválido de vias

### Identificador

O identificador único do caso de teste “Criação de uma rua com um número inválido de vias” é **C02**.

### Itens

Este caso se propõe a testar o módulo Interface, a fim de verificar se a rua é corretamente desenhada, e o módulo Dados do Mapa, para verificar se a informação inválida do número de vias não é obtida pelo sistema.

### Entradas e Saídas

A entrada para esse caso de teste é o uso da ferramenta que cria uma rua no mapa e um número de vias que não seja do tipo *inteiro*, ou seja, considerado inválido.

Como saída, o sistema não deve aceitar o valor do número de vias, e uma mensagem de erro deve ser produzida, além de desenhar corretamente a rua no mapa.

## Criação de uma rua conectada a outra

### Identificador

O identificador único do caso de teste “Criação de uma rua conectada a outra” é **C03**.

### Itens

Este caso se propõe a testar o módulo Interface, a fim de verificar se uma rua é corretamente conectada a outra no mapa.

### Entradas e Saídas

A entrada para esse caso de teste é o uso da ferramenta que cria uma rua, conectando-a a um dos nós de uma outra rua já existente no mapa.

Como saída, o sistema não deve desenhar corretamente a ligação entre as duas ruas.

## Remoção de uma rua

### Identificador

O identificador único do caso de teste “Remoção de uma rua” é **C04**.

### Itens

Este caso se propõe a testar o módulo Interface, a fim de verificar se a rua é corretamente removida do mapa, e o módulo Dados do Mapa, para verificar se as informações referentes à rua são removidas do sistema.

### Entradas e Saídas

A entrada para esse caso de teste é a remoção da rua desejada com a tecla *Delete*.

Como saída, o sistema deve remover corretamente a rua do mapa.

## Criação de um semáforo em uma rua com uma temporização válida

### Identificador

O identificador único do caso de teste “Criação de um semáforo em uma rua com uma temporização válida” é **C05**.

### Itens

Este caso se propõe a testar o módulo Interface, a fim de verificar se um semáforo é corretamente desenhado em uma rua, e o módulo Dados do Mapa, para verificar se a informação da temporização é obtida pelo sistema.

### Entradas e Saídas

A entrada para esse caso de teste é o uso da ferramenta que cria um semáforo em uma rua do mapa e uma temporização que seja um número válido (*inteiro* ou *de ponto flutuante*).

Como saída, o sistema deve aceitar o valor da temporização, sem apresentar uma mensagem de erro, além de desenhar corretamente o semáforo no mapa.

## Criação de um semáforo em uma rua com uma temporização inválida

### Identificador

O identificador único do caso de teste “Criação de um semáforo em uma rua com uma temporização inválida” é **C06**.

### Itens

Este caso se propõe a testar o módulo Interface, a fim de verificar se um semáforo é corretamente desenhado em uma rua, e o módulo Dados do Mapa, para verificar se a informação inválida da temporização não é obtida pelo sistema.

### Entradas e Saídas

A entrada para esse caso de teste é o uso da ferramenta que cria um semáforo em uma rua do mapa e uma temporização que não seja um número inteiro, ou seja, considerada inválida.

Como saída, o sistema não deve aceitar o valor da temporização, apresentando uma mensagem de erro , além de desenhar corretamente o semáforo no mapa.

Como saída, o sistema não deve aceitar o valor da temporização, apresentando uma mensagem de erro, que deverá instruir o usuário acerca do valor correto a ser fornecido a este parâmetro. A janela de definição da temporização será exibida novamente a seguir e a entrada de um valor válido permitirá que o sistema desenhe o semáforo no mapa.

## Remoção de um semáforo em uma rua

### Identificador

O identificador único do caso de teste “Remoção de um semáforo em uma rua” é **C07**.

### Itens

Este caso se propõe a testar o módulo Interface, a fim de verificar se o semáforo é corretamente removido do mapa, e o módulo Dados do Mapa, para verificar se as informações referentes ao semáforo são removidas do sistema.

### Entradas e Saídas

A entrada para esse caso de teste é a remoção do semáforo desejado com a tecla *Delete*.

Como saída, o sistema deve remover corretamente o semáforo do mapa.

## Criação de um novo mapa

### Identificador

O identificador único do caso de teste “Criação de um novo mapa” é **C08**.

### Itens

Este caso se propõe a testar o módulo Interface, a fim de verificar se o mapa é corretamente removido da área de desenho e simulação.

### Entradas e Saídas

A entrada para esse caso de teste é a ferramenta de criação de um novo mapa, acessada a partir do menu do aplicativo.

Como saída, o sistema deve remover corretamente o mapa previamente criado na interface, sem que haja a ocorrência de erros.

## Definição de uma nova velocidade máxima, válida, de uma rua

### Identificador

O identificador único do caso de teste “Definição de uma nova velocidade máxima, válida, de uma rua” é **C09**.

### Itens

Este caso se propõe a testar o módulo Dados do Mapa, para verificar se a nova velocidade máxima é corretamente alterada no sistema.

### Entradas e Saídas

A entrada para esse caso de teste é o acesso à ferramenta de alteração da velocidade máxima de uma rua e uma velocidade que seja um número válido (*inteiro* ou *de ponto flutuante*).

Como saída, o sistema deve alterar corretamente a velocidade máxima da rua desejada, ou seja, ela deve ser devidamente capturada sem que haja a ocorrência de erros.

## Definição de uma nova velocidade máxima, inválida, de uma rua

### Identificador

O identificador único do caso de teste “Definição de uma nova velocidade máxima, inválida, de uma rua” é **C10**.

### Itens

Este caso se propõe a testar o módulo Dados do Mapa, para verificar se a nova velocidade máxima não é alterada no sistema, por ser inválida.

### Entradas e Saídas

A entrada para esse caso de teste é o acesso à ferramenta de alteração da velocidade máxima de uma rua e uma velocidade que não seja um número, ou seja, considerada inválida.

Como saída, o sistema não deve alterar a velocidade máxima da rua desejada, apresentando uma mensagem de erro para o usuário.

## Definição de parâmetros gerais válidos

### Identificador

O identificador único do caso de teste “Definição de parâmetros gerais válidos” é **C11**.

### Itens

Este caso se propõe a testar o módulo Dados do Mapa, para verificar se os parâmetros gerais da simulação (fluxo de entrada, distância de segurança entre os veículos e passo da00207263148simulação) são corretamente alterados no sistema.

### Entradas e Saídas

A entrada para esse caso de teste é o acesso à ferramenta de alteração dos parâmetros gerais e parâmetros que sejam números, isto é, que são válidos.

Como saída, o sistema deve alterar corretamente os parâmetros, sem que haja a ocorrência de erros.

# Procedimentos de teste

## Procedimento 1

### Identificador

Oi!

### Finalidade

Indicar os casos de teste incluídos neste procedimento.

### Necessidades especiais

Necessidades especiais para a execução do procedimento.

### Ações

Ações necessárias para executar o procedimento: iniciar, executar, medir, suspender, recomeçar, parar ou concluir.

### Relatórios

Como serão apresentados e guardados os resultados dos testes (inclui um log).