**Sistema de Gerenciamento de Palestras**

**Roteiro de Teste**

**Disponibilidade**

## 1. Suíte de Teste: Disponibilidade

|  |  |
| --- | --- |
| **Caso de Teste: populateData** | |
| Objetivo do Teste: A proposta deste caso de teste é validar a criação de uma disponibilidade. | |
| Pré-condições: Acessar o sistema com usuário público. | |
| Ações do Passo: | Resultados Esperados: |
|  | 1. O método deve receber uma string contendo a disponibilidade por extenso. Os três primeiros caracteres devem identificar o dia da semana no formato SEG, TER, QUA, QUI, SEX, SAB, DOM.   2. Caso o sistema encontre uma data específica, deve adicionar à disponibilidade. Senão, adiciona apenas o horário.  3. A data deve ser composta de dia, mês e ano. Se o dia for informado como “00”, o sistema deve entender que a disponibilidade é para o dia da semana do mês/ano informado.  4. O horário disponível é composto de início e fim. O horário de início deve ser menor que o final.  5. Fim do caso de teste. |
|  |  |
| Fluxo: Principal | |
| Massa de teste:  "Ter, 10/11/2015, 10:00-12:00"  "Seg, 00/12/2015, 08:00-18:00"  "Dom, 12:00-14:00"  "Qua, 18/11/2015" | |

|  |  |
| --- | --- |
| **Caso de Teste: match** | |
| Objetivo do Teste:  A proposta deste caso de teste é comparar o tipo de disponibilidade. | |
| Pré-condições: Acessar o sistema com usuário público. Cadastrar uma disponibilidade através do método populateData. | |
| Ações do Passo: | Resultados Esperados: |
| 1. Acessar a funcionalidade através da interface **Disponibility** e acionar a opção **match**. | 1. O sistema recebe os parâmetros ano, mês, dia, dia da semana e duração.   2. O sistema compara com a disponibilidade existente. Caso sejam compatíveis,o sistema retorna o valor verdadeiro, senão retorna o valor falso. |
| Fluxo: FP + Alternativo | |