Relatório de Análise de Requisitos

Segue abaixo a análise de requisitos via ReqSCity, ferramenta essa desenvolvida num projeto PIBITI apoiado pela Capes/CNPQ.

Requisitos considerados ambíguos e X denota qual o motivo:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nº Requisito | Palavra Ambigua | Algoritmo Flexible Ambiguity | Ambiguidade Analitica | Ambiguidade por Coordenação | Ambiguidade de ligação |
| 8 | 🗴 | ✓ | ✓ | ✓ | ✓ |
| 9 | 🗴 | ✓ | 🗴 | ✓ | ✓ |
| 12 | ✓ | 🗴 | ✓ | ✓ | ✓ |
| 13 | ✓ | 🗴 | 🗴 | ✓ | ✓ |
| 14 | ✓ | 🗴 | 🗴 | ✓ | ✓ |
| 20 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 24 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 26 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 27 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 28 | 🗴 | ✓ | 🗴 | ✓ | ✓ |
| 29 | ✓ | 🗴 | ✓ | ✓ | ✓ |
| 31 | 🗴 | ✓ | 🗴 | ✓ | ✓ |
| 34 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 35 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 37 | ✓ | 🗴 | ✓ | ✓ | ✓ |
| 38 | 🗴 | ✓ | ✓ | ✓ | ✓ |
| 40 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 46 | 🗴 | ✓ | ✓ | ✓ | ✓ |
| 48 | ✓ | ✓ | ✓ | ✓ | 🗴 |
| 49 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 50 | ✓ | ✓ | 🗴 | ✓ | ✓ |
| 51 | ✓ | ✓ | 🗴 | ✓ | ✓ |

Requisitos considerados incompletos e X denota qual o motivo:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nº Requisito | Ausência de Verbos | Voz Passiva | Falta de Sujeito | Dummy Subject |
| 1 | ✓ | ✓ | 🗴 | ✓ |
| 2 | ✓ | ✓ | 🗴 | ✓ |
| 4 | ✓ | ✓ | 🗴 | ✓ |
| 5 | 🗴 | ✓ | ✓ | ✓ |
| 6 | ✓ | ✓ | 🗴 | ✓ |
| 10 | 🗴 | ✓ | ✓ | ✓ |
| 11 | 🗴 | ✓ | ✓ | ✓ |
| 12 | 🗴 | ✓ | ✓ | ✓ |
| 13 | 🗴 | ✓ | ✓ | ✓ |
| 14 | ✓ | 🗴 | ✓ | 🗴 |
| 15 | 🗴 | ✓ | ✓ | ✓ |
| 24 | ✓ | 🗴 | ✓ | 🗴 |
| 25 | ✓ | ✓ | 🗴 | ✓ |
| 26 | ✓ | 🗴 | ✓ | 🗴 |
| 28 | ✓ | 🗴 | ✓ | 🗴 |
| 29 | ✓ | 🗴 | ✓ | 🗴 |
| 31 | ✓ | 🗴 | ✓ | 🗴 |
| 32 | ✓ | ✓ | 🗴 | ✓ |
| 33 | ✓ | 🗴 | ✓ | 🗴 |
| 35 | ✓ | ✓ | 🗴 | ✓ |
| 36 | 🗴 | ✓ | ✓ | ✓ |
| 37 | 🗴 | ✓ | ✓ | ✓ |
| 39 | ✓ | 🗴 | ✓ | 🗴 |
| 41 | 🗴 | ✓ | ✓ | ✓ |
| 48 | ✓ | 🗴 | ✓ | 🗴 |
| 52 | ✓ | 🗴 | ✓ | 🗴 |

Requisitos Considerados Bons:

|  |
| --- |
| Nº do Requisito |
| 3 |
| 7 |
| 42 |
| 43 |
| 44 |
| 45 |
| 47 |
| 16 |
| 17 |
| 18 |
| 19 |
| 21 |
| 22 |
| 23 |
| 53 |
| 54 |
| 55 |
| 30 |

Requisitos Contextualizados:

|  |
| --- |
| Nº do Requisito |
| 5 |
| 10 |
| 11 |
| 12 |
| 14 |
| 15 |
| 22 |
| 23 |
| 26 |
| 30 |
| 31 |
| 34 |
| 35 |
| 37 |
| 38 |
| 39 |
| 40 |
| 41 |
| 42 |
| 45 |
| 46 |
| 49 |
| 51 |

Requisitos Contextualizados porém com o sentido do sensor incompleto:

|  |
| --- |
| Nº do Requisito |
| 10 |
| 11 |
| 15 |
| 34 |

Requisitos Contextualizados porém com o sentido do atuador incompleto:

|  |
| --- |
| Nº do Requisito |
| 15 |

Requisitos que foram tratados:

# RQ1 : Include Remedy List - In this requirement, the manager will insert the remedy list available in the system ;  
# RQ2 : Include Pharmacy List - In this requirement, the manager will insert the pharmacy list registered in the program ;  
# RQ3 : Consult Remedy in nearest pharmacies - In this requirement the user will have the possibility to consult a medication in the application ;  
# RQ4 : Send Warnings on Medicines - In this user case the user will receive notification of availability of medications at nearby pharmacies ;  
# RQ5 : Display traffic data, according to predefined rules ;  
# RQ6 : Take routes - In this requirement, the user will have the best route available from their location to the nearest pharmacy ;  
# RQ7 : The system will present a security policy that ensures the reliability of the information registered by users in the application ;  
# RQ8 : The system will present ease of use of its interface, providing the user with little effort to reach a goal ;  
# RQ9 : The system will present availability of information and connectivity, providing easy access to data related to medicines and nearest pharmacies registered according to their location ;  
# RQ10 : Collect sensor data ;  
# RQ11 : Analyze sensor information ;  
# RQ12 : Display traffic flow history graphically ;  
# RQ13 : Low response time for congestion detection ;  
# RQ14 : Low response time when the traffic light is activated ;  
# RQ15 : Management of sensor and actuator data ;  
# RQ16 : Dynamic change according to the context ;  
# RQ17 : The system should display on a map markers for crimes that occurred in downtown Chicago in the years of 2016, 2017, 2018 ;  
# RQ18 : The system should display on a map the markers related to the crimes that occurred in downtown Chicago at a interval specified by the user ;  
# RQ19 : The system should display the number of crimes that have occurred in a given time interval specified by the user ;  
# RQ20 : The system should provide information about a specific crime to the user ;  
# RQ21 : The system should list crimes of a type specified by the user ;  
# RQ22 : The system should display a bar graph with the number of the main crimes that occurred in the years 2016, 2017 and 2018. Optionally, the user can specify the interval, in months, that he wishes to consult ;  
# RQ23 : The system will only use data whose content is in the public domain ;  
# RQ24 : The system will be entirely WEB, the HTML markup language and the Javascript scripting language will be used. The Leaflet API will be used to display the map. The data necessary to meet the demands of the system must be obtained through the InterSCity platform ;  
# RQ25 : When launched, the application should open with the initial display, containing the developers names, UFMA and LSDI logo, application name and the button for the user to click and access the map ;  
# RQ26 : Bearing in mind that the system will use data from shared bikes in Paris ,France, and it will be developed for use in Brazil. In this specific case, points of the City of Paris will be correlated with the City University Dom Delgado - UFMA. The user location will be obtained automatically ;  
# RQ27 : The system should identify which bike station is closest to the user location, which has at least one free bike. There should be a marker, at the station location, in addition to a text box, containing the station name and the number of bikes available ;  
# RQ28 : The system must be connected to the internet to provide the necessary information to the end user ;  
# RQ29 : The system will be designed for the android platform ;  
# RQ30 : Configure the ContextNet Gateway IP and Port - As the system works based on the Client / Server paradigm it is necessary to previously configure the IP and Port of the gateway that will receive the data ;  
# RQ31 : Smart Meter needs to be connected and functional - Stores IP and Port using Android key value persistence mechanism ;  
# RQ32 : Set the UUID, from the intercity, of the resource ;  
# RQ33 : The system sends a request with the hello message to the gateway and the gateway is expected to send a response ;  
# RQ34 : The system reads the voltage, current and power information from the sensor ;  
# RQ35 : When a Processing Node receives an electrical consumption message, it must register it with InterSCity plataform;  
# RQ36 : The user must log in to the monitoring system ;  
# RQ37 : The system graphically displays information on the energy consumption in KWh of each building ;  
# RQ38 : The system constantly monitors the energy consumption of each building and alerts users if the building experiences energy consumption ;  
# RQ39 : The system allows messages to be sent to people inside the building ;  
# RQ40 : The system should receive data from the Chicago Data Portal website api and show the user the water temperature of a Chicago beach at a certain date and time ;  
# RQ41 : The system should monitor the turbidity of the water through the API data ;  
# RQ42 : The system should monitor the height of the waves using the API data ;  
# RQ43 : The application will have a map showing the location of the beaches through markers ;  
# RQ44 : The application will have a map showing the location of the beaches through markers. When selecting a marker, the user will obtain the beach information ;  
# RQ45 : The system must consume data from a REST API in JSON format ;  
# RQ46 : The system should perform data updates every 5 seconds ;  
# RQ47 : The system should be able to add new monitoring points dynamically ;  
# RQ48 : The occupancy rate for parking spaces will be displayed on the web page provided in percentage via a pie chart ;  
# RQ49 : The service will be saving the records of the highest and lowest historical occupancy rate of each month. These records can be consulted at any time and the data for the highest occupancy rate and the lowest occupancy rate will be displayed on the screen ;  
# RQ50 : The system will use the usability guide provided by the federal government of Brazil, highlighting the 7 guidelines addressed by the government ;  
# RQ51 : All code produced must conform to high-level models. Thus, changes in the application code must be reflected in changes in the high-level models. The top-level models are the UML diagrams of class, sequence and components. Therefore, the application structure must conform to the class diagrams, and the generated algorithms must conform to the sequence diagrams ;  
# RQ52 : Must be logged ;  
# RQ53 : The system must show a log of sensors ;  
# RQ54 : The user must be able to search ;  
# RQ55 : The system must detect ;