Documentation Procedures

*Project TraN-SiS*

We will follow the following procedures throughout the project to maintain uniformity and enable others to easily understanding the code written by a team member, finally making it to easy to complete the documentation of the software at the end of the project.

1. **Identifier Nomenclature** (for variables, objects, functions, structures, etc.)
   * *Identifiers* to have meaningful abbreviation (short forms) of names. (Avoid single letter names like ‘a’, ’b’,’ x’, ’y’ for important variables.)
   * The words in the *identifier* names must be in *lowercase* separated by *underscore* (‘\_’).
2. **Comments**
   * Make it compulsory to comment at every possible process (step) of the code. (Try to do this while you are writing the code and not after writing the code because it might sometimes be hard to remember the processes after finishing the code.)
   * It is mandatory to write a **case description** before defining the cases (in the form of a multiline comment). The cases may be in the form of *if-else* condition or in the form of *switch-case* conditions. The description will include the following:
     1. Every possible case condition.
     2. Process to be carried out in each case. (\*These comments can also be given just before the case or if-else condition rather than all at the beginning of all cases.)
   * It is mandatory to write a **function description** before defining the function (in the form of a multiline comment). The description will include the following:
     1. The description of the process being carried out in the function.
     2. The *parameters* being passed into the function during function call.
     3. *Return* *type* of the function. (\*If possible also describe the possible *return* values the function can give.)
3. **Flowcharts**
   * (\*If possible) Before writing any code try to make a *flowchart*. This would be the ultimate key to master the understanding of the program. And it will be an important tool to explain the code to others too.
4. **Function Description**
   * As each of the major functions will be in a separate file as a **module** ,also try to make a document file(\*.docx) along with each of the functions which contains the *flowchart* and the *function description*(which can be extracted from the *comments* in the code).
5. **Debugging Code**
   * Make a debugging code (which would generally be a function) for the code that you make.
   * The *debugging function* would generally print values of prominent or important variables/objects.
6. **Name/ID**:
   * ***Names*** of an object or variable do not have any specifications when user defined otherwise if the names are automatically generated they would be the same as the *ID* of the object.
   * Each ***ID*** of an object or variable would be of the form:
     1. **<structure-code><object number>.<sub-object code>**
     2. Each Structure will have a character code which will be used as the “*structure-code*”. The following table contains the code for each structure :

|  |  |  |
| --- | --- | --- |
| **Structure Name** | **Identifier** | **Structure Code** |
| Map Extension Node | map\_extn | MEN |
| Traffic Signal Junction | traffic\_sig\_jn | TSJ |
| Lane | lane | LAN |
| Street | street | STR |
| Traffic Light | traffic\_light | TSL |
| Vehicle | vehicle | VEH |
| Route | route | ROU |

* + 1. Object number will be a **4 digit number** which will be according to the number of objects that are created.
    2. Sub-object code will be there for the objects that are created within a structure. Its format will be same as the above.
    3. Example 1: If the 105th Vehicle is created it will have the ID : **VEH0105**.
    4. Example 2:If the 5th Traffic Light is created in the 21st Traffic Signal Junction it will have the ID : **TSJ0021.TSL0005.**