COMP 3004 – *SCAPES* Assignment #1

Due: Tuesday, October 1, 2019 at 4:00 PM (afternoon)

Collaboration: This assignment must be completed **individually**

Functional Model Document (partial)

Table of content:

1. Functional requirements
2. Non-functional requirements
3. Use cases

Document content:

Functional requirements:

- Provide a description of every functional requirement for the entire *SCAPES* system, as found in the *SCAPES* project description
- All requirements must be uniquely numbered for traceability

Non-functional requirements:

- Provide a description of every non-functional requirement for the entire *SCAPES* system
- Non-functional requirements must be organized in the nine (9) categories presented in class (section 2.2, slides #9 and 10)
- There must be at least two (2) different requirements from each of the nine (9) categories
  - You must cover all the requirements that are present in the *SCAPES* project description
  - You will have to make up some reasonable requirements in order to cover all categories
  - Remember: NFRs must be precise and verifiable
- All requirements must be uniquely numbered for traceability

Use cases:

- UML use case diagram:
  - You must provide **one** UML use case diagram that shows the entire *SCAPES* system
  - The set of use cases must cover all the functional requirements
  - The use case diagram must show the system boundaries, all actors and their relationships with the use cases
  - The use case diagram must include at least two (2) high-level use cases for more abstract and grouped functionality, as well as detailed use cases with refinements, with the correct relationships between them
  - The use case diagram must include at least three (3) different error use cases, with the correct relationships to the originating use cases
- Use case table-based descriptions:
  - You must provide the table-based descriptions for all the high-level and detailed use cases involving a user compiling and running an existing SCAPL program
  - The level of detail in the breakdown of use cases must match the ARENA case study from the textbook
  - Each use case table-based description must include the use case id, use case name, participating actors, flow of events, entry and exit conditions, quality requirements, and traceability
  - You must provide the table description for at least one (1) error use case that can occur in compiling and running a SCAPL program

- Traceability:
  - All use cases must be uniquely numbered for traceability
  - Every use case must must be linked to one or more requirements for traceability

Grading

Grading breakdown:

- *Functional Model Document:*

Functional requirements	22%
Non-functional requirements	18%
Use cases diagram	30%
Use case tables	30%

Grading criteria:

- *Document completeness*: All the material must be present; completeness covers both breadth and depth; for example, it is insufficient to only present high-level use cases, they must be accompanied by detailed ones.
- *Document correctness*: All the material must be presented accurately, with appropriate diagrams and in the correct format; superfluous, unnecessary material is considered incorrect.
- *Document traceability*: Every part of every model should be numbered and traced to each other, using a numbering scheme established in the requirements section.
- *Document presentation*: Professional-level documentation is expected, subject to a 10 mark deduction as penalty.

Format

Documentation deliverables must be submitted as a **PDF document**. They must be typed and legible, and they must be **professional**, including a cover page, page numbers, as well as section numbers and names. All UML diagrams must be produced using a drawing tool, and not hand-drawn. Documents that do not conform to these specifications will not be graded.