**Document 05 – Sprint 1 Plan**

This document is contained in your GitHub repository in a folder named *docs*. It must be there by the due date provided on the Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Group |  | | Group Member Names |  |
|  | |  |  | | --- | --- | | 1. |  | | 2. |  | | |  |  | | --- | --- | | 3. |  | | 4. |  | |
|  |  |  |

The objective of this sprint is to produce a working system that implements a subset of the requirements that provide the most value to a scientist.

1. **Sprint 1 Backlog**

|  |
| --- |
| **Deliverable**  Choose a set of requirements that (a) you can accomplish in the form of a working system by the end of the sprint, (b) provide the most value to a scientist. List them in the order of priority in the tables below. Key:  Requirement – the requirement number from Document 02 – Project Description & Requirements  Short Title – come up with a short title that is unique  Description – Copy from Document 2. When you learn new information about the requirement as you go along, add/edit it to this section or the “Notes” section. |

|  |  |
| --- | --- |
| Priority | 1 |
| Requirement |  |
| Short Title |  |
| Description |  |
| Notes |  |

|  |  |
| --- | --- |
| Priority | 2 |
| Requirement |  |
| Short Title |  |
| Description |  |
| Notes |  |

…

1. **Class Diagram**

|  |
| --- |
| **Deliverable**  Consider at least several of the requirements from above. Develop a neat, legible, properly sized UML class diagram(s) showing your initial design. Notes:   * Classes should show the most important state and behavior. * Don’t show getters. * Don’t show any state or behaviors for Gui/UI class(es), just the class itself. If you feel something is important and/or it helps you, then do add state and behavior. * You may need to introduce some classes that interface between the Gui and domain classes, so show those if you identify them. If not you will discover them when you start coding. * It is not important to capture every detail. Every time you look at your document, you will see new things you didn’t think of before: a required instance variable, a parameter for a method, a missing method, a method in the wrong class, a new class, etc. **The important point is to establish a starting point for the design**. * This can be drawn by hand. You may find software that will allow you to draw the diagram, but you must make sure it uses the exact format covered in class. * You must use proper UML notation. * You should spend 1-2 hours on this, and then maybe ½ an hour to produce a final diagram. |

List requirements considered:

<<remove-class diagram goes here>>

1. **Prototype**

|  |
| --- |
| **Deliverable**  Consider at least several of the requirements from above. Sketch your Gui or UI. Notes:   * This may be one drawing, or it may be several. * There are prototyping tools you can use, or you can draw it by hand. This is one, but I haven’t used it. If you find it useful, and you discover any tips/tricks, please share them with me (or the class) asap (if you want to): <http://pencil.evolus.vn/> |

List requirements considered:

<<remove-prototype image goes here>>

1. **Task Identification**

|  |
| --- |
| **Deliverable**  Consider one or more requirements from above. Break the work down into small tasks, assign to group members and enter into your Sprint 1 Project in GitHub. See: *Document 03 – Development Requirements*, Section 1. |

**Start coding!**