1. **Team Name: Sleepy Ducks**
2. **Team Leader for this deliverable: Lorenzo Gomez**
3. **Team Members: Anton Ryjov, Gemuele Aludino, Lorenzo Gomez**
4. **Meetings:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Time-date** | **Attendees** | **Agenda** | **Action Items (who will do what)** |
| 10-20-2019 (Sunday)  17:30 - 20:30 | Lorenzo, Anton,  Gem | Discussing/reviewing specifications, and writing the proposed system | Lorenzo  wrote introduction  Anton  discussion of SRS  Gem  discussion of SRS |
| 10-21-2019  (Monday)  16:40 - 20:00 | Lorenzo, Anton,  Gem | Preliminary work on  systems  diagram  (part I) | Lorenzo  systems diagram work on paper  Gem  creation of Visual Paradigm document on computer |
| 10-22-2019 (Tuesday)  16:40 - 20:00 | Lorenzo, Anton,  Gem | Preliminary work on  systems  diagram  (part II) | Lorenzo  systems diagram work on paper  (revisions)  Gem  creation of Visual Paradigm document on computer (revisions) |
| 10-23-2019 (Wednesday)  16:40 - 21:30 | Lorenzo, Anton, Gem | Finished work on SRS  documentation | Lorenzo  classified functional/non-functional requirements  Anton  classified functional/non-functional requirements; use-case documents  Gem  created team report document |

1. **Weekly Time Logs:**

|  |  |  |
| --- | --- | --- |
| **Person** | **Total Time in minutes** | **Tasks** |
| Lorenzo | 670 | whiteboard brainstorming; SRS documentation |
| Anton | 670 | whiteboard brainstorming; use-case documents |
| Gem | 670 | writing/typing team report |
| **Total Time:** | 2010 |  |

1. **Issues:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Issue Number** | **Discovery Date** | **Resolution Date ( Est. – Act. )** | **Responsible Person** | **Description ( Prob / Resolution )** |
| Functional vs. non-functional requirements | 10-23-2019 | 10-23-2019 | Anton | We had issues decoupling the system into the different use cases. |

1. **Files and repository locations:**

|  |  |  |
| --- | --- | --- |
| **Filename** | **Location** | **Contents** |
| SRS.docx | <https://github.com/thebigG/Tasker.git> | SRS deliverable |

1. **Plans for Coming Week:**

* Discussion of a uniform design language for Tasker
* Further acquaintance with Qt and its APIs, Qt Creator, as well as C++ in general

1. **Comments:** *a paragraph from each engineer describing what they have done/learned from this deliverable*

**Engineer 1:** *Lorenzo Gomez*

*Thinking more deeply about system models and use cases forced us to start decoupling our system into its moving parts. This was challenging because we had to re-visit our initial specification and peel the functional and non-functional layers of Tasker. We realized that we had more non-functional requirements than we thought as our system has a specific target—unix-based systems. Our design philosophy is also geared towards making things simple for the user, which made us not only re-visit our specification, but we also had to explicitly state these quality constraints(non-functional requirements) in our specifications and use cases.*

**Engineer 2:** *Gemuele (Gem) Aludino*

*I had a few challenges understanding the difference between functional and non-functional requirements, but thanks to the discussions between Lorenzo and Anton — they were able to clear up some of my misunderstandings and I have a much clearer idea between the two. I find that non-functional requirements have more to do with product philosophy, as opposed to descriptions of functionality — it is in the name, after all.*

*The Visual Paradigm software had a bit of a learning curve to it, but after some time, I was able to get the gist of using it, and I am looking forward to completing more of our assignments with it.*

**Engineer 3:** *Anton Ryjov*

*Visual Paradigm was kinda a breeze for use cases, quite similar in a few ways to UMLet (open-source uml maker made by some students) but obviously significantly cleaner and more professional. Didn’t have problems with it. We discussed most of the things in front of a whiteboard which was very helpful in a few cases.*