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)** |
| 11-04-2019  (Monday)  16:40 – 19:00 | Lorenzo, Anton,  Gem | Sketch core entity classes | Lorenzo  recognize entity classes(commitment, Task, Session, etc) and discuss their roles  Anton  recognize entity classes(commitment, Task, Session, etc) and discuss their roles  Gem  recognize entity classes(commitment, Task, Session, etc) and discuss their roles |
| 11-05-2019  (Tuesday)  18 – 20:00 | Lorenzo, Anton,  Gem | Sketch core utility classes | Lorenzo  recognize utility class and its role for presentation layer(stats)  Gem  recognize utility class and its role for presentation layer(stats)  Anton  recognize utility class and its role for  presentation layer(stats) |
| 11-06-2019 (Wednesday)  16:40 - 19:00 | Lorenzo, Anton,  Gem | Discuss classes that will persist on disk(user commitments data) | Lorenzo  Sketch data-persistent classes and variable naming convention for Commitment Window  Gem  Sketch data-persistent classes and variable naming convention for Stats Window  Anton  Sketch data-persistent classes and  variable naming convention for  TimerWindow |

1. **Weekly Time Logs:**

|  |  |  |
| --- | --- | --- |
| **Person** | **Total Time in minutes** | **Tasks** |
| Lorenzo | 600 | Work on CreateCommitment window |
| Anton | 600 | Work on Stats and commitments view |
| Gem | 600 | Work on Timer Window |
| **Total Time:** | 1800 |  |

1. **Issues:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Issue Number** | **Discovery Date** | **Resolution Date ( Est. – Act. )** | **Responsible Person** | **Description ( Prob / Resolution )** |
| Abstracting Listeners | 11-07-2019 | 11-11-2019 | Lorenzo,Anton,Gem | On our Thursday meeting, we realized that abstracting complexity is HARD. We want our listener code to be flexible enough, but also well-define for users to be able to interface easily with them. |

1. **Files and repository locations:**

|  |  |  |
| --- | --- | --- |
| **Filename** | **Location** | **Contents** |
| Tasker.pro | [https://github.com/thebigG/Tasker.git](https://github.com/thebigG/Tasker.gitQt) | QT Project file |

1. **Plans for Coming Week:**

* Refactor code to satisfy naming conventions
* Fix alignment between all windows
* Discuss an abstract class model for Tasker

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

**Engineer 1:** *Lorenzo Gomez*

*Since we had never used QT, getting acquainted with it in just a week was definitely a challenge. So far it has proven to be quite useful, so we hope that it will suit all of our needs in the future.*

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

*I really enjoy designing GUIs, and had a blast working on my section (Widget\_CommStats.\*) – things are coming along, and after today’s presentation, my outlook on the project overall is positive. We just have to straighten a few kinks, in terms of naming conventions from within our source files, and get any labels/boxes/widgets perfectly sized and aligned (I am usually the one to notice if something is “off”), but that’s no big deal.*

**Engineer 3:** *Anton Ryjov*