

| **1. APT Project Advancement Summary** |
| --- |
| A continuación, encontrarás distintos campos que deberás completar con la información solicitada. |

| APT Project Advancement Summary | Different deliveries and sprints have been completed. Currently, an initial release or MVP has been delivered, as well as a second release or increment. Work is currently underway on Sprint 6, which is part of Release 3.  Up until now different User Stories have been delivered and there’s only been slow downs during sprint 4 as detailed later.  Las principales funcionalidades trabajadas durante estos release están relacionadas con la integración con servicios de Odoo, almacenar y administrar proyectos, realizar una clusterización para saber las horas a utilizar por proyecto, gestionar los parámetros de la página, entre otros.  The main functionalities developed during these releases are related to integrating the program with Odoo services like, storing and managing projects, clustering of projects to know the work hours to use for each, managing the page’s parameters, etc. |
| --- | --- |
| Objectives | There were no changes to the project’s main objective. |
| Methodology | There were no changes to the Methodology used. |
| Evidence of progress | Github(Source code): The program itself, fully working. We consider this to be the most important evidence as it lets us check the actual amount of progress made to the project and verify if it’s working correctly.  Database (Relational Model): The Database Model to use for the program. We consider it relevant in this case as it makes it possible to verify that a database is being used and that the software can indeed store and realize different operations in it.  Backlog (User Stories): Stories used to develop the system. We consider them relevance as they show the functionalities the project has to ultimately have, allowing to validate the final functionalities with those originally planned.  Architecture (Technical Specifications Diagram): In this case it’s a graphic image of the final architecture to be used. In this case it lets people have a summary of how the project works at a glance. This is considered evidence as it can be used to validate that the program works with the interfaces and connections used.  In this case we’ve maintained the project’s quality thanks to following some of Python’s standards, seeking to use variables with clear names, system testing before delivery, etc. |

| **2. Work Plan Monitoring** |
| --- |
| Carefully examine your work plan, mainly focusing on the columns of advancements and adjustments. |

| Work Plan | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Competence or Competence Units | Activities | Resources | Activity Duration | Responsible[[1]](#footnote-0) | Observations | Progress State | Adjustment |
| Database and Virtual environments management.  Develop a Software able to systemize processes.  Build data models.  Make quality assurance testing.  Implement integral systematic solutions. | Different sprints with the goal to deliver value are considered for this software’s development. Some relevant activities for this project are:   1. Project Clustering Development 2. Project Management 3. User Management 4. Parameters Management | Computers, Database. | Tasks will be advanced and developed during different sprints, so each doesn’t have a duration in hours but must be completed during a certain sprint.   1. Clustering: Sprints 2-6 2. Project Management: Sprints 1-4 3. User Management: Sprints 1-4 4. Parameter Management: Sprints 3-4 | Due to the nature of Agile Methodology, there’s no defined or specific roles for each team member, as it is considered for the team to be self-organizing in how they approach their tasks. | At this moment in time the biggest problems that we have encountered and may happen again are mostly related to the team’s lack of communication with the client and knowledge. | 1. **Clustering:** In Progress 2. **Project Management:** Completed 3. **User Management:** Completed 4. **Parameters Management:** In Progress | *There’s been no changes to the project’s scope nor its functionalities.* |

| **3. Monitoring Adjustments** |
| --- |
| Go in depth about the observations of your work plan. Analyze all activities planned and point out which aspects helped and which became obstacles to the plan’s execution. Explain how you came over or will come over those obstacles. Finally, point out the adjustments you’ll make to the work plan from this analysis. |

| Factors that eased or hindered the development of my work plan:  During this project’s development the team has had some factors that helped us make it though the difficulties that may affect the project negatively, some of these are:  \*Team Organization: The team has had a good organization of both tasks and external responsibilities, for example, documentation, making development go by with almost no time drawbacks.  \*Team Communication: Communication has been essential to avoid problems when developing the program, presenting ideas and errors, avoiding delays in the project and having a good work environment. \*Tools used: Using development tools that the team had mostly a hold on helped development greatly.  Even with all these factors that helped the development, there were some factors that produced delays which we’re still trying to mitigate as much as possible, some of these factors are:   \*Lacking communication with the 2nd team: During development there were some problems when trying to communicate with the other team who were in charge of developing functionalities of great importance for the project, hindering the organization between both teams.  \*Lacking communication with the client: During development there were some problems with continuous communication with the client, which brought problems when presenting results and advancements, as these aren’t always liked by them.  \*User stories estimation: There’s been problems when estimating the difficulty of/ the time needed to complete some stories, giving them a lesser difficulty and scoring than they should’ve actually had.  \*Lack of knowledge and experience: Development has shown the lack of knowledge and experience the team has when working in a more professional environment with this methodology and the processes used which resulted in some delays during development. |
| --- |

| Activities adjusted or deleted:  Only redundant stories of lesser importance have been deleted up until now, nothing essential to the project. |
| --- |

| Delayed and pending activities: We’ve only had to delay two stories towards our next sprint (sprint 5), these are the following:  \*”As the admin I need the Log to be automatically deleted after some time to avoid database overload”.  \*”As the admin i need the time to be a parameter i can modify as needed”  These stories were planned for sprint 4, but due to a lack of knowledge on how to make an automatic process in django and making it able to be modified via parameters inputted by the user, we decided to move them to sprint number 5, we also decided to lower the priority of both, as the teacher gave us some high priority integration functions, asking for us to center our efforts on those tasks first. |
| --- |

1. In case the APT Project is being done in groups, these columns must indicate those responsible for each activity. This will later allow us to divide the evaluation for each team member. [↑](#footnote-ref-0)