}

**Guide 1. Definition of APT Project**

**Capstone Course**

1. **PART I**

|  |
| --- |
| **1. Personal Background** |

|  |  |
| --- | --- |
| Student Name | **Diego Ozzy Silva Herrera - Fabián Cristóbal Valenzuela Antúnez** |
| Tax ID | **21,617,475-5 / 21,456,004-6** |
| Field of Study | **Computer Engineering** |
| Location | **Padre Alonso de Ovalle (PAO)** |

|  |
| --- |
| **2. APT Project Description** |

|  |  |
| --- | --- |
| Project Name | Automated Inventory Manager for a Pet Store |
| Area(s) of performance | * Full-Stack Web Development * Databases * Process Automation |
| Skills | * Develop software solutions in accordance with organizational requirements. * Build scalable and normalized data models. * Program queries and routines for information management. * Implement comprehensive solutions to optimize business processes. * Manage IT projects using agile methodologies. |

|  |
| --- |
| **3. APT Project Justification** |

|  |  |
| --- | --- |
| Relevance of the APT Project | The project seeks to solve the problem faced by many small and medium-sized businesses, in this case a pet store, with regard to inventory management. Manual stock management often leads to human error, wasted time, and difficulties in obtaining real-time visibility of available products.  We chose this topic because it is highly relevant to the field of computer science, where digital transformation and internal process optimization represents strategic value for businesses. The project is set in the context of a small store located in Chile, but it can be extrapolated to any retail business. It directly impacts business managers, who require accurate information to avoid stock shortages or oversupply, and indirectly benefits customers by ensuring product availability.  The value of this project lies in providing a local web solution, accessible within the store, that facilitates data-driven decision-making, reduces human error, and increases operational efficiency. |
| APT Project Description | The project consists of creating an automated web-based inventory management system. It will enable:   * Register products and update stock in real time. * Process sales and purchases with automatic inventory adjustment. * Generate basic reports on stock levels and best-selling products.  The solution will be implemented in **Python and Django,** with **SQLite** as the database engine, and will be developed in **a local environment** during the semester.   The solution will be implemented in Python and Django, with SQLite or PostgreSQL as the database engine, and will be worked on in a local environment during the semester. |
| Relevance of the project to the graduate profile | This project is directly linked to the **graduate profile of the Computer Engineer**, since:   * **Specialized skills**:   + Modeling and managing databases.   + Developing software solutions applying best practices.   + Managing IT projects within limited timeframes and resources.   + Performing validation and quality control tests. * **Generic skills**:   + Written and oral communication (documentation + presentation).   + Teamwork and leadership.   + Innovation and entrepreneurship.  This project requires the integration of these skills to solve a real need for automation in a business context. |
| Relationship with professional interests | Our professional interests focus on **Full-Stack software development**, applying backend, frontend, and database development, and building solutions that solve real problems in business environments. This project reflects these interests by allowing us to work with modern web development tools (Django, HTML, CSS, JS) and databases, such as SQLite, bringing both areas together.  The experience of carrying out this APT will allow us to strengthen key technical skills, consolidate professional competencies, and bring us closer to real management contexts, directly contributing to our future professional development. |
| Feasibility of developing the APT project | The project is feasible within the academic framework for the following reasons:   1. **Semester length**: 18 weeks, sufficient time to plan, develop, and present. 2. **Course hours**: the assigned working hours allow progress to be made on development milestones. 3. **Required materials**: personal notebook, free software (Python, Django, DB). 4. **External factors that facilitate** previous experience in programming and databases. 5. **External factors that hinder** limited time; this will be addressed with detailed planning (Gantt chart). |

1. **PART II**

|  |
| --- |
| **4. Objectives** |

|  |  |
| --- | --- |
| General objective | Develop an automated inventory management web system for a pet store that optimizes stock control, sales, and purchases, and generates reports to support decision-making. |
| Specific objectives | * Design and model the database to record products, sales, and inventory movements. * Implement a backend module in Django to perform CRUD operations. * Develop an intuitive web interface that allows interaction with the system. * Automate inventory updates with each sale or purchase recorded. * Generate basic reports that provide visibility into inventory status. * Validate the quality of the system through unit and comprehensive testing. |

|  |
| --- |
| **5. Methodology** |

|  |
| --- |
| Description of the Methodology |
| The **agile Scru** methodology will be applied, with partial deliveries (sprints) and constant feedback.  **Main stages**:   1. Definition of requirements and data modeling. 2. System architecture design. 3. Backend development and initial testing. 4. Frontend development and integration testing. 5. Final validation and presentation preparation.   **Team roles**:   * **Diego Silva**: backend development and database. * **Fabián Valenzuela**: frontend development and integration. * Both share responsibility for planning, presentation, technical documentation, evidence, and testing. |

|  |
| --- |
| **6. Evidence** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Type of evidence**  **(preliminary or final)** | **Name of evidence** | **Description** | **Justification** |
| **Preliminary** | **Data model** | **Diagrams and database structure** | **Reflects initial analysis and technical planning** |
| **Progress** | **Interface prototype** | **Sketches and initial views of the system** | **Evidence of design and usability** |
| **Final** | **Functional system** | **CRUD for products, sales, and reports** | **Achievement of the overall objective** |
| **Final** | **Testing and validation** | **Unit and comprehensive test records** | **Ensures software quality and reliability** |
| **Final** | **Technical documentation** | **Final report and user manual** | **Demonstrate professional application and skills** |

|  |
| --- |
| **7. Work Plan** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **APT Project Work Plan** | | | | | | |
| Competency or competency units | Name of Activities/Tasks | Description of Activities/Tasks | Resources | Duration of activity | Responsible[[1]](#footnote-1) | Comments |
| **Documentation** | **Report documentation** | **Required documentation and evidence** | **Word, GitHub** | **2 weeks** | **Both** | **Academic requirement** |
| ***Database modeling*** | **Table and relationship design** | **Database modeling and relationships** | **Draw.io** | **1 week** | **Diego** | **Project foundation** |
| ***Backend programming*** | **CRUD for products and movements** | **CRUD development and inventory logic** | **Django, Python** | **3 weeks** | **Diego** | **Integrates with frontend** |
| **Frontend programming** | **User interfaces** | **Web interfaces and usability** | **HTML, CSS, Django templates** | **3 weeks** | **Fabian** | **Must be usable and intuitive** |
| **Integration** | **Backend–frontend connection** | **Connection of all system logic with the user interface** | **Django ORM** | **2 weeks** | **Both** | **Requires joint testing** |
| **Reports** | **Basic reporting module** |  | **Django + libraries** | **2 weeks** | **Both** | **Important for added value** |
| **Testing** | **Unit and integration testing** | **Final report and oral presentation** | **Pytest, logs** | **2 weeks** | **Both** | **Quality assurance** |

|  |
| --- |
| **8. Gantt chart** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activity** | **Phase 1** | | | | **Phase 2** | | | | | | | | | | | | **Phase 3** | | | |
| **S 1** | **S 2** | **S 3** | **S 4** | **S 5** | **S 6** | **S 7** | **S 8** | **S 9** | **S 10** | **S 11** | **S 12** | **S 13** | **S 14** | **S 15** | **S 16** | | **S 17** | **S 18** |
| **Report Documentation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **Table and Relationship Design** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **CRUD for products and movements** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **User interfaces** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **Backend/frontend connection** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **Basic reporting module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **Unit and integration testing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **Final presentation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |

1. If the APT Project is a group project, the names of those responsible for each task or activity should be indicated in this column. This will subsequently allow for individual assessment of each member. [↑](#footnote-ref-1)