

# Prácticas supervisadas en el desarrollo de un software de seguimiento de la salud de personas con Enfermedad de Parkinson

Brizuela Elías, Mamani Villca, Sonia,  
Raggio Franco, Rodríguez Jonathan.

## Abstract

*Presentamos los resultados preliminares de las prácticas supervisadas que se vienen realizando desde el año 2020 en el proyecto Tecnologías de software para el seguimiento de actividades y terapias grupales de bienestar en personas con Enfermedad de Parkinson (Telepark). Este proyecto se orienta al diseño y desarrollo de un software que tiene el fin de acompañar la atención de personas con Enfermedad de Parkinson tomando como modelo un grupo de soporte físico y social de la región. Los temas desarrollados son los siguientes, de los cuales se describen resultados preliminares y líneas de avance: 1. Análisis y diseño del sistema de teleseguimiento en actividades y terapias de bienestar en personas con EP, 2. Construcción del backend de módulos gestión de personas con EP y ficha médica de la aplicación, 3. Construcción del módulo de gestión de ficha médico-clínica y la clasificación de actividades de la aplicación Telepark y 4. Construcción del frontend del módulo gestión de persona con EP de la aplicación. Los resultados preliminares permiten señalar el logro de nuevas herramientas de procedimiento y aprendizajes conceptuales y de actitudes favorables al trabajo en equipo intragrupo, intergrupos de investigación dentro del GIDAS e interclaustró. Se propone continuar avanzando en el desarrollo del sistema Telepark y concretar una prueba piloto a futuro.*

## Palabras Clave

Práctica supervisada, Tecnología de software, Enfermedad Parkinson

## 1. Introducción y objetivo

La Enfermedad de Parkinson (EP) constituye una condición de salud multisistémica clasificada tradicionalmente dentro de los trastornos del movimiento. Se caracteriza por presentar síntomas motores, no motores y premotores. Su diagnóstico es clínico y su causa es desconocida, presumiendo un origen ecogenético.

Constituye la segunda enfermedad neurodegenerativa (la primera es la Enfermedad de Alzheimer) en prevalencia poblacional, aumentando la prevalencia a medida que aumenta la edad del grupo etario [1].

La EP requiere de un abordaje interdisciplinario a fin de atender las distintas manifestaciones de la enfermedad que generan cambios, en general indeseados, en la vida de las personas con Parkinson y su entorno. Acompañar el proceso de cambio y promover una adaptación activa y saludable es la misión de las ciencias de la salud a este respecto. Considerando que los desarrollos tecnológico-informáticos actuales permiten optimizar los procesos de acompañamiento a través del desarrollo de sistemas inteligentes, las ciencias de la computación vienen a constituirse en un pilar importante del abordaje interdisciplinario al que nos referimos.

A principios de 2020 comenzó la inquietud de profesionales de la salud y miembros de la Universidad Tecnológica Nacional (UTN) Facultad Regional La Plata sobre el desarrollo de instancias informáticas para el acompañamiento de las personas con EP.

Al mismo tiempo, la declaración de pandemia y aislamiento social subsecuente llevó a que los estudiantes próximos a egresar de la UTN, se encontraran con dificultades a la hora de realizar sus prácticas supervisadas (PS) en empresas y otros espacios de desarrollo habitual.

De este modo, se construyó un grupo interdisciplinario orientado al desarrollo de un software de soporte de la salud en el área

de la EP donde estudiantes avanzados pudieran llevar adelante sus prácticas supervisadas. Así se inició el proyecto Telepark, con sede en el Grupo de investigación y desarrollo aplicado a sistemas informáticos y computacionales (GIDAS) de la UTN y con la participación de miembros de la Universidad Nacional de La Plata, con el objetivo de construir una tecnología de software aplicado al ámbito de salud y bienestar de las personas con EP. Se buscó automatizar un conjunto de tareas habituales de asistencia y acompañamiento de salud, permitiendo ofrecer indicadores para la toma de decisiones y con el potencial de favorecer el seguimiento de la salud de las personas con EP.

El software cuyos resultados preliminares presentamos tiene su blanco comunitario en un grupo de soporte físico y social interdisciplinario que se desarrolla con personas con EP. Como el inicio del proyecto Telepark coincidió con la pandemia por Covid-19, las actividades se vienen desarrollando en forma virtual. Los estudiantes avanzados se han incorporado mediante convocatorias abiertas. Describiremos en este artículo las actividades realizadas hasta el momento con sus resultados preliminares.

Estas actividades se agrupan en los siguientes tramos: 1) capacitación sobre EP, 2) investigación del estado del arte, 3) armado del plan de proyecto, 4) definición de metodologías y tecnología a aplicar, 4) capacitación en herramientas, 5) ejecución del proceso de desarrollo.

Los temas centrales de las PS son los siguientes:

1. Análisis y diseño del sistema de teleseguimiento en actividades y terapias de bienestar en personas con EP. Este constituye el tema introductorio, donde se propuso realizar actividades de análisis y diseño del primer módulo de la aplicación Telepark, consistente en la gestión de la ficha personal administrativa y la gestión de ficha médica de las personas con EP.

2. Construcción *backend* [9] de módulos gestión de personas con EP y ficha médica

de la aplicación. En este punto se propuso la construcción del *backend* para los módulos mencionados, lo que implicó la creación de la *application programming interface* (API) para el alta, baja y modificación de las entidades, así como su conexión a una base de datos que permitiera su persistencia a largo plazo. Asimismo, el tema incluyó el desarrollo de un módulo de usuarios para el control de acceso a la información, el despliegue del *backend* en un entorno que permitiera su uso, el testeado, refinamiento en la integración con el *frontend* [9], versionado y la documentación.

3. Construcción del módulo de gestión de ficha médico-clínica y la clasificación de actividades de la aplicación Telepark. El módulo de gestión de ficha médico-clínica abarcó la gestión de diagnósticos, la de indicación de medicamentos, la de obra social y la de la evolución de cada persona con EP.

4. Construcción *frontend* de módulos gestión de persona con EP de la aplicación. En este tema se esperaba avanzar en el desarrollo del módulo conteniendo gestión de persona con EP con sus datos personales, la gestión de domicilio y el módulo de gestión de eventos referido a los sucesos que pudieran acontecer no previsto en el seguimiento inicial de la persona con EP (por ejemplo, ausencia en una actividad de recreación o entrevista de evolución).

## 2. Metodología

La industria del software prevé diversos recursos: metodologías, técnicas, herramientas, paradigmas, heurísticas, entornos y lenguajes de programación, estándares y catálogos de buenas prácticas para llevar adelante todas las etapas del proceso de desarrollo del software y su administración bajo un estricto enfoque ingenieril [12],[11],[4].

Inicialmente se realizó una revisión bibliográfica del estado del arte del dominio y entrevistas a profesionales de la salud entendidos en el tema, esto último para

avanzar en la etapa de elicitación de requerimientos.

Para el logro de resultados elegimos a Rational Unified Process (RUP) [5] como proceso de desarrollo de software iterativo; para modelar las diferentes vistas en Lenguaje Unificado de Modelado (UML) [2] y [6] con la herramienta Enterprise Architect (<https://sparxsystems.com/>) y para la gestión del proyecto la herramienta Trello (<https://www.trello.com/>).

La elicitación de requerimientos comenzó con ciclos de charlas, indagación y estudio de los rasgos fundamentales de EP apoyados en técnicas de entrevistas e intervenciones grupales a través de lluvia de ideas [10]. En esta etapa se tuvo en cuenta la aplicación de recursos gestionados correcta y eficientemente, con la asistencia de instrumentos propuestos por Loucopoulos y Karakostas en 1995 [8].

Para el desarrollo de los módulos mencionados en los temas de las PS (cfr. sección Introducción y objetivo) se implementaron diferentes tecnologías aplicando una arquitectura *model-view-controller* (MVC) [6].

En el *frontend* se implementó React (<https://es.reactjs.org/>) que es una biblioteca de JavaScript para crear interfaces de usuario interactivas de forma sencilla, junto con Redux (<https://es.redux.js.org/>) para el manejo del estado de la aplicación web.

Esta decisión arquitectónica facilitó la integración del *frontend* con el *backend*. Ambos, *frontend* y *backend*, utilizaron una autenticación basada en credenciales de usuarios para comunicarse entre sí. Además de esto, se utilizó un sistema de perfiles para determinar el acceso a datos sensibles más allá de la autenticación (autorización).

En el *backend* se implementó Python 3 (<https://www.python.org/downloads>) utilizando Django Rest Framework (<https://www.django-rest-framework.org/>) para construir la API Rest, de donde se consumirán los datos. Junto a esta tecnología se construyó la base de datos relacional, en motor MySQL (<https://www.mysql.com/>), una herramienta de código abierto y segura.

El uso de Python con el *framework* Django, se decidió por los beneficios que aporta en el manejo de datos a futuro. Tener resueltos varios de los aspectos requeridos para el *backend*, facilita el desarrollo y permite solucionar rápidamente aspectos como el manejo de usuarios, conexión a base de datos y la implementación de los distintos *endpoints* que la aplicación requiere.

Para la gestión del versionado del *frontend* se creó un repositorio en GitHub (<https://github.com/>) y para el *backend* se creó un repositorio en GitLab (<https://gitlab.com/>), donde cada uno de los participantes del proyecto va incorporando el producto de su desarrollo.

Cabe destacar que estos desarrollos continúan en el presente y que lo que presentamos en la siguiente sección son resultados preliminares.

### 3. Resultados

#### 3.1. Revisión sistemática

De la revisión sistemática de antecedentes - a inicios de 2020- se seleccionaron dos artículos como los más significativos para el desarrollo de las PS en el contexto del proyecto Telepark: [7] Linares-del Rey et al, 2019 y [3] García Vázquez et al, 2013.

El primer antecedente mencionado constituye una exploración de aplicaciones móviles de soporte que tomó fuentes primarias en inglés o español del período 2011-2016 que presentasen, analizasen o validasen un sistema basado en una aplicación (app) con utilidad o diseño específico para la EP. Los autores identificaron 125 aplicaciones: 56 se clasificaron con potencial utilidad en la EP y 69 con un diseño específico para la EP. Estas últimas se subclasificaron en cuatro grupos: 23 apps de información sobre EP (que ofrecían información acerca de la EP para profesionales, pacientes, familiares y cuidadores), 29 apps de valoración (que permitían testear la evolución de los signos de la EP mediante la marcha, el equilibrio,

el temblor, el habla y otros aspectos del comportamiento), 13 apps de tratamiento (que ofrecían al profesional y al paciente una serie de pautas sobre el tratamiento farmacológico o no farmacológico, de rehabilitación física y similar) y 4 apps que vinculaban valoración y tratamiento. Los autores de este trabajo concluyeron que existía un gran número de aplicaciones móviles con potencial utilidad, pero la evidencia científica acerca de los mismos era escasa y de baja calidad.

El segundo antecedente mencionado detalló el diseño y evaluación de un servicio de *e-salud* orientado a la estimulación y seguimiento de personas con EP que tuvieran algún trastorno cognitivo. En el trabajo se identificaron algunas aplicaciones de la familia de los sistemas conocidos como *brain games*, con creciente popularidad y múltiples plataformas tecnológicas para los usuarios. En su artículo, los autores informaron el desarrollo y evaluación de una aplicación móvil de teleseguimiento y estimulación cognitiva cuyos usuarios eran personas con EP. Según los autores, la utilidad de este sistema podría extenderse a otras enfermedades, permitiendo al terapeuta hacer el seguimiento de la actividad de rehabilitación extramuros del paciente. Los antecedentes revisados sirvieron para conocer que existía un corpus de información en desarrollo, a la vez de hallar el espacio de vacancia en el cual podría incluirse el sistema Telepark.

### 3.2. Desarrollo del sistema

Se planteó la puesta en marcha de un proceso de desarrollo de software según lineamientos de ciclo de vida del sistema expuestos en [4], juntamente con un proceso de investigación de tecnologías disponibles para aplicar o refinar en el presente proyecto.

#### 3.2.1. Diagramas funcionales

Se elaboró el *diagrama de paquetes* que se presenta en Figura 1:

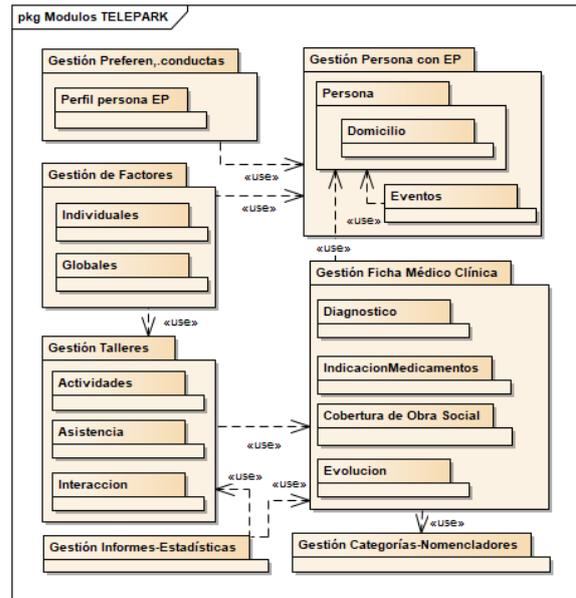


Figura 1 -Diagrama de paquetes - Arquitectura global del sistema Telepark.

También se modeló el *diagrama de clases* que se expone a continuación en Figura 2:

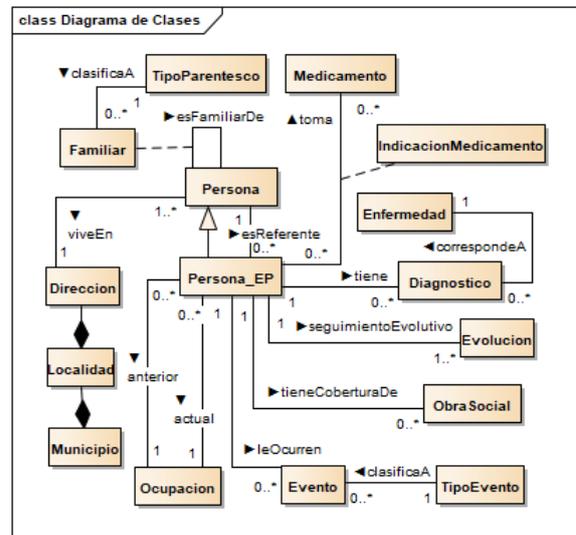


Figura 2 -Diagrama de Clases de Dominio del sistema Telepark.

#### 3.2.2. Desarrollo del backend y frontend

Al día de hoy tenemos un *backend* con varios de los módulos y operaciones funcionando. El mismo permite la consulta, modificación y eliminación de distintas entidades, persistencia en la base de datos,

autenticación y autorización ante las operaciones solicitadas a la API.

Se ha probado, además, la integración con una versión del *frontend*, pudiendo realizar operaciones de manera exitosa, que utiliza el *backend* para obtener distintos datos necesarios para el formulario, y el guardado en la base de datos de la entidad resultante.

Se gestionaron las cuentas en GitHub donde el código ya se encuentra disponible para los miembros del proyecto, ya sea para su consulta, uso, o modificación. Esto también pensando a futuro, para una integración más favorable al realizar una publicación en los distintos ambientes, ya sean de testing o producción.

A continuación, en Figura 3 se muestra una captura de árbol del framework:

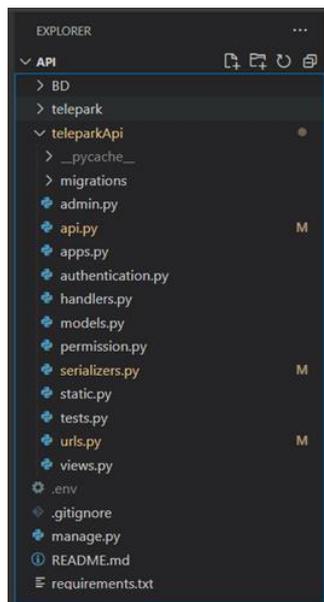


Figura 3 - Árbol de la estructura de la aplicación.

Dentro de este contexto, se ha logrado la realización del formulario donde se da de alta al paciente con EP. El desarrollo del lado del *frontend* llevó a cabo las interfaces de la carga de datos del paciente como también los datos del referente familiar o acompañante o cuidador de la persona con Parkinson. Este proceso finalizó con la integración del formulario unido al *backend* y realización de la interfaz de logeo por donde el usuario va a poder ingresar al sistema, cargar datos y llevar a cabo consultas.

Asimismo, dentro de referido contexto, se ha logrado construir la estructura de la base de datos correspondiente a los módulos diagramados y desarrollar las interfaces correspondientes al módulo de *gestión de diagnósticos*, donde el usuario podrá verificar la lista de diagnósticos o síndromes que podría asignarse a cada persona con EP (como comorbilidad) e ingresar, modificar o eliminar los datos correspondientes a cada registro como el nombre de la enfermedad y la fecha del diagnóstico.

A continuación, en Figura 4 se muestra la GUI de la gestión del referente de la persona con EP:



Figura 4 – Ejemplo de GUI de la aplicación.

Cabe destacar que para estos logros se realizaron interconsultas con otros grupos de investigación del GIDAS para revisar la viabilidad de lo construido y su correcta implementación, resultando en mejoras en cuanto a la infraestructura y fiabilidad de la solución en cuestión. No obstante, esta solución se encuentra en un ciclo constante de refinamiento que permite mejorar la eficiencia, facilidad de uso, fiabilidad o rapidez de la misma.

#### 4. Conclusión

Los resultados preliminares permiten señalar la importancia que ha tenido la generación de prácticas supervisadas en el grupo de trabajo Telepark del GIDAS, permitiendo no sólo la adquisición de nuevas herramientas de procedimiento y aprendizajes conceptuales de los estudiantes próximos a graduarse, sino también el desarrollo de actitudes favorables al trabajo en equipo

intragrupo y la colaboración recíproca entre grupos de investigación, tanto de pares como entre profesores y estudiantes.

Se propone continuar avanzando en el desarrollo del sistema Telepark y concretar una prueba piloto en el año 2022, cuando se espera que las actividades del grupo de soporte físico y social destinatario retome sus actividades presenciales, si el desarrollo de la pandemia lo permite.

### Agradecimientos

A los responsables del GIDAS y del grupo Telepark y a los integrantes del grupo de soporte físico y social que sirve de modelo para este trabajo.

### Referencias

[1] Balestrino, R., & Schapira, A. H. V. (2020). Parkinson disease. *European Journal of Neurology*, 27(1), 27-42. <https://doi.org/10.1111/ENE.14108>

[2] Booch, G., Rumbaugh, J., & Jacobson, I. (1999). *The Unified Modeling Language User Guide*. Addison-Wesley. Reading.

[3] García Vázquez, C., Moreno Martínez, E., Valero Duboy, M. A., Martínez Juez, M. T., & Torre Calero, M. S. (2013). Servicio ubicuo de estimulación cognitiva orientado a personas con enfermedad de Parkinson. En: *XI Jornadas de Ingeniería Telemática (JITEL 2013)* (pp. 273-280). Granada 28-30 octubre, 2013. ISBN 978-84-616-5597-7. Recuperado desde <http://oa.upm.es/26053/>

[4] ISO/IEC/IEEE 12207:2017(E) Systems and software engineering - Software life cycle processes.

[5] Jacobson I., Booch G., & Rumbaugh J. (2000). *El Proceso Unificado de Desarrollo de Software*. Addison-Wesley.

[6] Larman, C. (2000). *UML y Patrones. Una introducción al análisis y diseño orientado a objetos y al proceso unificado*. Prentice Hall.

[7] Linares-del Rey, M., Vela-Desojo, L., & Cano-de la Cuerda, R. (2019). Aplicaciones móviles en la enfermedad de Parkinson: una revisión sistemática. *Neurología*, 34(1), 38-54. <https://doi.org/10.1016/j.nrl.2017.03.006>

[8] Loucopoulos, P. & Karakostas, V. (1995). *System requirements engineering*. McGraw-Hill.

[9] Platzi.com Business and Legal Affairs (Accedido el 4/9/2021) <https://platzi.com/blog/que-es-frontend-y-backend/>

[10] Pohl, K. (2010). *Requirements engineering: fundamentals, principles, and techniques*. Springer Publishing Company, Incorporated.

[11] Pressman, R. S. (2010). *Software Engineering: A Practitioner's Approach*, McGraw-Hill.

[12] Sommerville, I. (2011). *Software engineering*. 9th Edition. Addison-Wesley, Pearson.