



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INFORMÁTICA
Departamento de Tecnologías y Sistemas de la Información

ANTEPROYECTO DEL TRABAJO FIN DE GRADO
GRADO EN INGENIERÍA INFORMÁTICA
TECNOLOGÍA ESPECÍFICA DE INGENIERÍA DE COMPUTADORES

**Diseño de un sistema de control de riego
automatizado aplicado a la viticultura**

Autor: Cristina Bolaños Peño
Director: Félix Jesús Villanueva Molina

Diciembre, 2018

Índice

| | |
|--------------------------------------|----------|
| 1. Introducción | 2 |
| 2. Tecnología Específica | 4 |
| 3. Objetivos | 5 |
| 4. Métodos y Fases de Trabajo | 6 |
| 5. Medios | 7 |
| 5.1. Medios Hardware | 7 |
| 5.2. Medios Software | 7 |
| 6. Referencias | 8 |

1. Introducción

El calentamiento global, la contaminación y la deforestación son conceptos plenamente conocidos y, lamentablemente, constantemente visibles en los medios de actualidad. Todos ellos afectan gravemente a las reservas de agua potable, o de uso en cultivos, del planeta, y por tanto, a la esperanza de vida humana.

En Castilla-La Mancha, nuestra tierra, 7.946.198 hectáreas de terreno son dedicadas al cultivo, y sólo en 474.910 de ellas se despliegan viñedos, sin tener en cuenta las diferentes variedades de uva que los agricultores manchegos pueden o no cultivar (Ministerio de Agricultura, Pesca y Alimentación, 2017). Para aprovisionar estos terrenos, se necesita un gran volumen de agua, el cual puede proceder de embalses, aguas depuradas, o acuíferos. Sin embargo, y precedida por los factores perjudiciales para el medio ambiente mencionados al inicio de este documento, la gestión de agua ante eventos extremos, por ejemplo sequías, será cada vez más importante.

En la actualidad, el sector agrario emplea distintas técnicas de riego para el suministro de agua. Según el Instituto Nacional de Estadística, 2016, el agua utilizada para riego en España se despliega en las siguientes opciones:

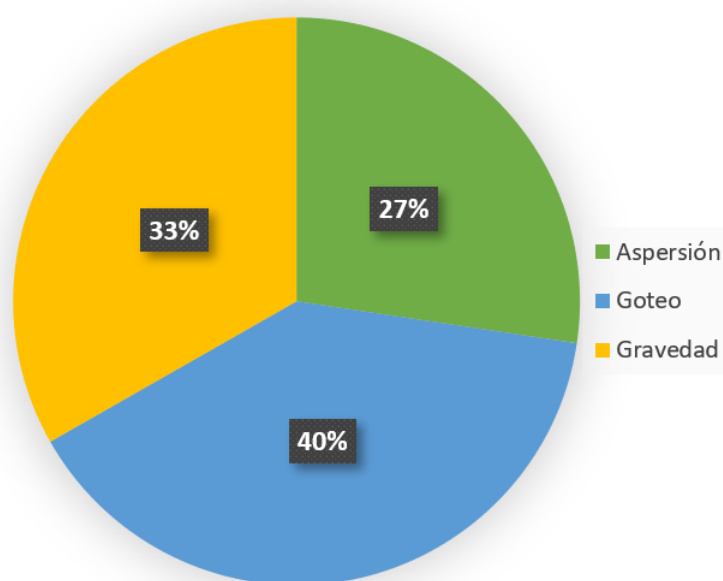


Figura 1: Sistemas de riego utilizados en España.

Actualmente se encuentran varias soluciones en el mercado como la aspersión, el riego por goteo o sistemas de riego automáticos dentro de invernaderos para el suministro de agua. Sin embargo, todos éstos funcionan dentro de franjas horarias sin ningún tipo de mecanismo de adaptación al terreno o a las condiciones que lo envuelven.

Es por ello que, con el fin de reducir al máximo posible el consumo de agua, con razón de cuidado de plantaciones y cultivos, se propone con este proyecto alcanzar una nueva alternativa adaptativa frente a los sistemas de riego convencionales. Para ello, se diseñará e implementará un sistema de control de riego automatizado para porciones de terreno grandes.

La solución optada se compone de dos subsistemas diferenciados:

1. Sensores y controladores del caudal del agua (**SubLC**): Estos componentes tendrán la tarea de monitorizar los distintos factores medibles del entorno de la plantación (temperatura, humedad del suelo, etc.) y controlar el caudal necesitado por el propio terreno en función de estos.
2. Servicio en la red (**SubI**): Se dispondrá un servicio disponible en la red que mostrará los datos recogidos por SubLC al usuario, y además controlará al sistema SubLC.

Se explorará más a fondo la arquitectura del sistema en el capítulo 3 del presente documento.

2. Tecnología Específica

En la Tabla 1 puede observarse la intensificación (tecnología) cursada y en la Tabla 2 las distintas competencias de la misma y abordadas en este proyecto:

Tabla 1: Intensificaciones de Ingeniería Informática, UCLM.

| Marcar la tecnología cursada | |
|------------------------------|-------------------------------|
| | Tecnologías de la Información |
| | Computación |
| | Ingeniería del Software |
| X | Ingeniería de Computadores |

Tabla 2: Justificación de las competencias abordadas en el proyecto.

| Competencia | Justificación |
|--|--|
| Capacidad de analizar y evaluar arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, así como desarrollar y optimizar software para las mismas. | En el proyecto se contempla el uso y diseño de sistemas distribuidos, por lo tanto es necesario evaluar distintas arquitecturas distribuidas para garantizar su eficiencia. |
| Capacidad de diseñar e implementar software de sistema y de comunicaciones. | Para este proyecto se diseñará el software de comunicaciones entre los distintos componentes y subsistemas. |
| Capacidad para analizar, evaluar, seleccionar y configurar plataformas hardware para el desarrollo y ejecución de aplicaciones y servicios informáticos. | Es necesario estudiar, evaluar y analizar distintas plataformas hardware para la sensorización y control del regadío del viñedo, las comunicaciones de los nodos y el procesamiento. |

3. Objetivos

El proyecto toma base en la siguiente arquitectura:

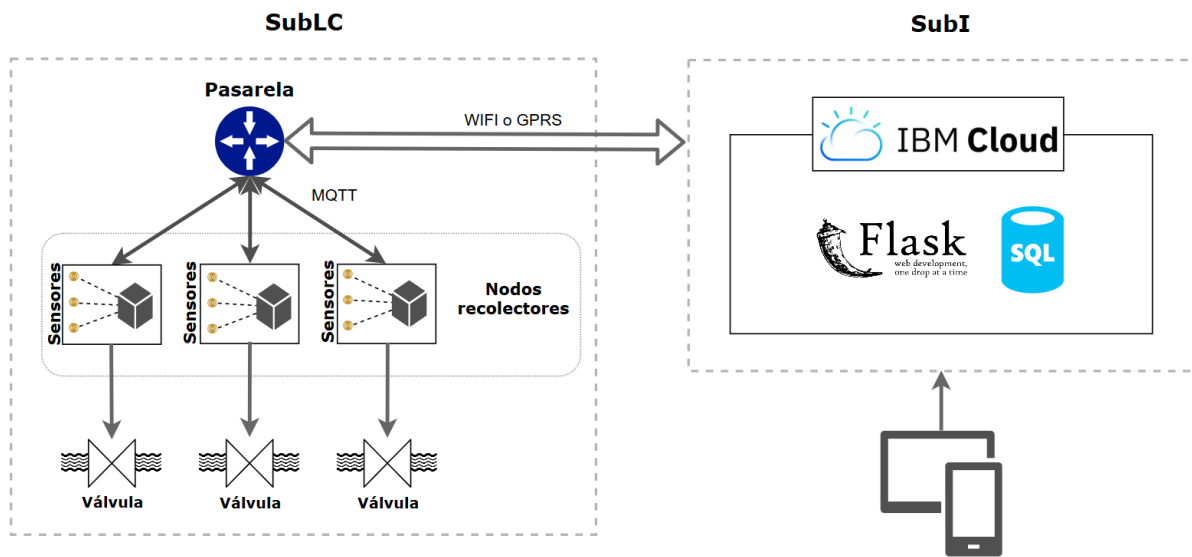


Figura 2: Arquitectura del sistema.

En la figura 2 podemos ver cómo, el usuario, desde cualquier dispositivo accedería a un servicio *cloud* y así conectar y/o controlar el sistema. Como ya se ha mencionado previamente, y reflejado en la propia ilustración, vemos los dos subsistemas del proyecto, los cuales se describirán a continuación:

SubLC Éste es el núcleo del proyecto. Se compone de 3 elementos fundamentales:

- **Nodo recolector:** Controlador con acceso a datos del entorno proporcionados por sensores. Para que el sistema funcione correctamente debe de haber varios de estos dispositivos repartidos por el terreno.
- **Válvula:** Sistema de control del caudal de agua de una específica sección del terreno. Debe ser eléctricamente manejable para poder ajustar su medida (electroválvula).
- **Pasarela:** Controlador que, como su propio nombre indica, actúa de pasarela entre los nodos recolectores y el subsistema SubI.

SubI Controla al subsistema SubLC y actúa como interfaz entre éste y el usuario. Además tiene la tarea de almacenar y analizar los datos recogidos. Se encontraría alojado en la nube.

4. Métodos y Fases de Trabajo

Para la realización de este proyecto se empleará el Proceso Unificado de Desarrollo, en adelante referido como PUD, como metodología de planificación y trabajo. Se ha optado por dicha alternativa por su relativa sencillez, generalidad, y la amplitud de control de software que ofrece.

Para diseñar y realizar los modelos de cualquier sistema, PUD hace uso del *Lenguaje Unificado de Modelado* (Object Management Group, 2017), en adelante referido como UML, el cual se ha utilizado en este proyecto en sus diferentes etapas. Cabe mencionar que PUD se caracteriza por estar **dirigido por casos de uso** y **centrado en la arquitectura**, mientras se trata de forma **iterativa e incremental**.

Por lo tanto, y siguiendo la estructura del ciclo de vida de un producto software definido por PUD, se han establecido las siguientes pautas o fases generales a seguir:

1. **Inicio:** Se llevará a cabo una descripción del producto final que se desea conseguir, estudiando el alcance del proyecto, su viabilidad, y la planificación del desarrollo del proyecto. Se obtienen:
 - a) Modelo de casos de uso, el cual describe las funcionalidades del sistema.
2. **Elaboración:** Se desarrollará y trabajará en el modelo obtenido en la fase anterior profundizando así en la arquitectura del sistema, consiguiendo la línea base de ésta. Se obtienen:
 - a) Modelo de análisis, con el que desarrollamos las funcionalidades en procesos, interfaces o bancos de datos.
 - b) Modelo de diseño, obteniendo un despliegue mucho más específico del modelo anterior.
3. **Construcción:** Se implementará y probará el producto, tanto su despliegue físico como el software de cada uno de sus componentes. Se obtiene:
 - a) Modelo de implementación, con el que se conseguirá representar la ejecución total del sistema.
 - b) Modelo de pruebas, el cual contiene el conjunto de casos de pruebas unitarias que se realizan al finalizar cada una de las fases del PUD.
4. **Transición:** Al llegar a esta etapa, el producto está en su versión beta y se procede a la evaluación del mismo en busca de errores o deficiencias. Se obtiene:
 - a) Versión del producto final.
 - b) Documentación del desarrollo.

5. Medios

5.1. Medios Hardware

Los medios hardware elegidos para desarrollar la solución aportada en este proyecto son:

- *Open Garden Outdoor Kit* (Open Garden, 2018): Se ha elegido este kit para cultivos de exteriores por la sencillez de su diseño y la gran capacidad de resolver nuestras necesidades de datos. Los componentes más relevantes del presente son:
 - ◇ *Open Garden Shield for Arduino* (pasarela del SubLC, en figura 2).
 - ◇ *Open Garden Node Board* (controlador del nodo recolector en SubLC, en figura 2).
 - ◇ DHT22 de *Open Garden* como sensor de humedad y temperatura.
 - ◇ Sensor de humedad del terreno de *Open Garden*.
 - ◇ Electroválvula de Open Garden, puede verse en [la tienda online de Open Garden](#).
- Lenovo Z50-70 (Lenovo, s.f.), como medio para la programación y configuración de los diferentes dispositivos.

5.2. Medios Software

Los medios software elegidos para desarrollar la solución aportada en este proyecto son:

- MQTT (IBM, 2014): Empleado para la comunicación entre la pasarela del SubLC con los nodos de los que dispongamos.
- IBM Cloud (IBM, 2018): Proveedor elegido para el alojamiento del subsistema SubI debido a su compromiso de colaboración con la UCLM y el acceso a sus herramientas para sus estudiantes.
- Flask (Ronacher, 2018): Utilizado para la creación de aplicaciones web basadas en Python (Python Software Foundation, 2018).
- SQL (ISO, 2018): Lenguaje de bases de datos utilizado para el almacenaje en el entorno *cloud* de aquellos que sean pertinentes.
- LaTeX (L^AT_EX3 Project Team, 2015): Necesario para la realización de la memoria y del presente documento.
- Bitbucket (Atlassian, 2018): Utilizado para el seguimiento de versiones del producto y de su documentación. Emplea Mercurial (O’Sullivan, 2015) para su uso.

6. Referencias

- L^AT_EX3 Project Team. (2015). L^AT_EX2e. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <https://www.latex-project.org/help/documentation/usrguide.pdf>
- Atlassian. (2018). Bitbucket. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <https://confluence.atlassian.com/get-started-with-bitbucket/get-started-with-bitbucket-cloud-856845168.html>
- IBM. (2014). MQTT v3.1 Protocol Specification. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <http://public.dhe.ibm.com/software/dw/webservices/ws-mqtt/mqtt-v3r1.html>
- IBM. (2018). IBM Cloud. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <https://www.ibm.com/cloud/>
- Instituto Nacional de Estadística. (2016). Volumen de agua usado por técnica de riego. Accedida por última vez el 15 de Noviembre de 2018. Recuperado desde https://www.ine.es/dyngs/INEbase/es/operacion.htm?c=Estadistica_C&cid=1254736176839&menu=ultiDatos&idp=1254735976602
- ISO. (2018). ISO/IES 9075-1:2008 Databases language – SQL. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <https://www.iso.org/standard/45498.html>
- Lenovo. (s.f.). Lenovo z50 Laptop. Accedida por última vez el 21 de Diciembre de 2018. Recuperado desde <https://www.lenovo.com/us/en/laptops/lenovo/z-series/z50/>
- Ministerio de Agricultura, Pesca y Alimentación. (2017). Encuesta sobre Superficies y Rendimientos de Cultivos. Accedida por última vez el 15 de Noviembre de 2018. Recuperado desde <https://www.mapa.gob.es/es/estadistica/temas/estadisticas-agrarias/agricultura/esyrce/>
- Object Management Group. (2017). Unified Modeling Language Specification Version 2.5.1. Accedida por última vez el 22 de Noviembre de 2018. Recuperado desde <https://www.omg.org/spec/UML>
- Open Garden. (2018). Open Garden Outdoor Kit. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <https://www.cooking-hacks.com/open-garden-outdoor-1node-1gw>
- O’Sullivan, B. (2015). Mercurial. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <https://book.mercurial-scm.org/read/>
- Python Software Foundation. (2018). Python. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <https://docs.python.org/2/tutorial/>
- Ronacher, A. (2018). Flask Documentation. Accedida por última vez el 13 de Diciembre de 2018. Recuperado desde <http://flask.pocoo.org/docs/1.0/>