

POLITECNICO DI TORINO  
Repository ISTITUZIONALE

Sistema de inmersión temporal: automatización de propagación in vitro de plantas utilizando herramientas libres

*Original*

Sistema de inmersión temporal: automatización de propagación in vitro de plantas utilizando herramientas libres / Orellano, Julio; Dellagiovanna, Agustín; Imanishi, Leandro; Mazzone, Virginia; Faedo, Nicolas. - (2016). (Intervento presentato al convegno AADECA 2016).

*Availability:*

This version is available at: 11583/2988069 since: 2024-04-24T11:46:24Z

*Publisher:*

Asociación Argentina de Control Automático (AADECA)

*Published*

DOI:

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/319469702>

# Sistema de Inmersión Temporal: Automatización de Propagación IN VITRO de Plantas Utilizando Herramientas Libres

Conference Paper · January 2017

CITATION

1

READS

4,288

5 authors, including:



**Nicolás Faedo**

Politecnico di Torino

60 PUBLICATIONS 578 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Wave energy control systems [View project](#)



DESTINY - Moment-based energy-maximising optimal control of wave energy systems [View project](#)

**SISTEMA DE INMERSIÓN TEMPORAL:  
AUTOMATIZACIÓN DE PROPAGACIÓN *IN VITRO*  
DE PLANTAS UTILIZANDO HERRAMIENTAS  
LIBRES**

**Julio Orellano\* Agustín Dellagiovanna\*  
Leandro Imanishi\* Virginia Mazzone\* Nicolás E. Faedo\***

*\* IACI - Departamento de Ciencia y Tecnología - Universidad  
Nacional de Quilmes*

**Resumen:** Este trabajo presenta el desarrollo de un Sistema de Inmersión Temporal y un sistema SCADA, ambos realizados con herramientas de Software Libre y Hardware Abierto, para un laboratorio de la Universidad Nacional de Quilmes. El diseño del sistema cuenta con biorreactores BIT integrados a un rack aislado lumínicamente del ambiente dentro del cual es posible el control del fotoperíodo. Además cuenta con un sistema neumático capaz de suministrar aire a las distintas secciones del rack. El sistema SCADA se compone de un servidor web conectado a un microcontrolador el cual se encarga del sensado y la actuación del sistema.

**Palabras Claves:** SCADA, Sistemas de Inmersión Temporal, Cultivo in vitro

## 1. INTRODUCCIÓN

La biotecnología se refiere a toda aplicación tecnológica que utilice sistemas biológicos y organismos vivos o sus derivados para la creación o modificación de procesos o productos para usos específicos (Nations, 1992). Entre estas tecnologías el cultivo y la propagación *in vitro* o micropropagación han sido de las primeras y más ampliamente difundidas aplicaciones prácticas de la biotecnología en plantas. El cultivo *in vitro* de plantas comprende un conjunto de técnicas que emplean células, tejidos u órganos vegetales, a través de las cuales se busca inducir la desdiferenciación de las células parentales en células meristemáticas (o embriogénicas), que luego se dividen y diferencian en nuevos órganos vegetales y/o plantas completas (Etienne and Berthouly, 2002).

En el laboratorio de Bioquímica, Microbiología e Interacciones Biológicas en el Suelo (LBMIBS) de la Universidad Nacional de Quilmes se llevan a cabo estudios básicos sobre la simbiosis fijadora de nitrógeno, particularmente en el arbusto actinorrhizo *Discaria trinervis*. La importancia

de la fijación biológica de nitrógeno a nivel ecológico y económico ha incentivado la investigación científica en el tema en los últimos 50 años (Werner *et al.*, 2014; Imanishi *et al.*, 2011).

La principal actividad del laboratorio LBMIBS implica la propagación y conservación del material vegetal, *Discaria trinervis*. Desafortunadamente no es posible la propagación sexual por semilla en condiciones de laboratorio. En este contexto la propagación *in vitro* surge como una alternativa a los métodos de propagación tradicionales. Éste implica el uso intensivo de recursos humanos para las diferentes etapas del proceso de propagación, especialmente en la etapa donde el material vegetal aumenta de manera exponencial su número. A su vez, las diferentes etapas del proceso de propagación *in vitro* requieren de diferentes condiciones de cultivo (e.g., temperatura, luz, reguladores de crecimiento, nutrientes).

Los Sistemas de Inmersión Temporal (SIT) desarrollados en los últimos años (Etienne and Berthouly, 2002; Georgiev *et al.*, 2014) abren la posibilidad de automatizar algunas de las etapas del cultivo *in vitro* convencional, optimizando enormemente el proceso de propagación. A la

reducción de los costos de mano de obra que significa la incorporación de esta tecnología se suman una mayor homogeneidad de las plantas propagadas y la posibilidad de controlar las condiciones de cultivo y ajustarlas de acuerdo a las necesidades particulares de cada etapa de la propagación *in vitro* y del experimento que se desee llevar a cabo. El monitoreo en línea de las variables relevantes para dicho proceso hacen que se eviten posibles accidentes y así pérdidas de cultivos.

Este trabajo describe el desarrollo de un SIT para satisfacer las necesidades del LBMIBS, tanto el diseño y la implementación mecánica, electrónica y neumática, como la utilización de software libre para la programación de un Sistema de Supervisión, Control y Adquisición de Datos (SCADA). Estos sistemas abarcan desde la transferencia de datos entre un servidor central y un número finito de Unidades de Acceso Remoto (RTU) hasta la comunicación con los controladores activos en planta, reúne información de proceso, transfiere al servidor central y proporciona datos de relevancia hacia las RTU (System, 2004).

## 2. SISTEMAS DE INMERSIÓN TEMPORAL

Los SIT son sistemas semi-automatizados de cultivo *in vitro*, basados en ciclos alternados de inmersión de las plantas en medio nutritivo líquido, seguido del drenado y la exposición de las plantas a un ambiente gaseoso (Georgiev *et al.*, 2014). El éxito en su uso radica principalmente en el régimen de inmersiones a los que se someten las plantas, siendo esencial controlar variables tales como el tiempo que las plantas estarán en contacto con el medio de cultivo, así como la frecuencia con las que estas inmersiones tienen lugar. Existen diversos modelos de SIT, mayormente basados en los diseños originales de Teisson y Alvard (RITA®: Recipiente para Inmersión Temporal Automatizada) y en el sistema desarrollado por Escalona y colaboradores (BIT®: Biorreactor de Inmersión Temporal). En el caso de este trabajo se opta por desarrollar un sistema de tipo BIT (Fig. 1). Los compartimentos que contienen las plantas y el medio de cultivo se encuentran en envases separados, y el pasaje del medio líquido de uno a otro compartimento tiene lugar a través de un tubo de silicona o vidrio que los interconecta. La inyección de aire comprimido en el frasco que contiene el medio de cultivo fuerza el pasaje del líquido hacia el segundo frasco, sumergiendo las plantas. El proceso luego se invierte mediante la aplicación de aire comprimido en el frasco con las plantas, retornando el sistema a su estado inicial (Escalona *et al.*, 1999).

La Fig. 2 muestra una fotografía del sistema, donde se pueden apreciar las dos secciones superiores destinadas a contener los biorreactores y el

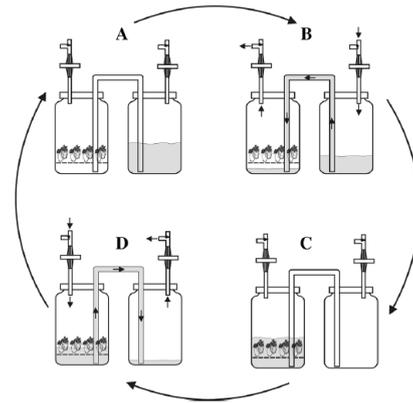


Fig. 1. Representación esquemática del funcionamiento del sistema de inmersión temporal BIT®.

piso inferior destinado a los sistemas eléctricos y neumáticos implementados para la automatización del proceso.



Fig. 2. Sistema de Inmersión Temporal desarrollado.

El rack se encuentra construido en acero SAE 1010 DD pintado con pintura epoxi para protección contra agentes oxidantes. Cada biorreactor está compuesto por dos frascos cerrados herméticamente, uno empleado para contener las plantas y otro en el que se encuentra el medio de cultivo, conectados entre sí a través de una manguera de silicona.

Para poder realizar el intercambio de medio de cultivo se genera diferencia de presión entre ambos recipientes mediante el ingreso de aire a través de una manguera unida a un filtro plástico con una membrana de PTFE (politetrafluoretileno) de 0.22 [ $\mu\text{m}$ ] de diámetro de poro, empleado para la esterilización del aire.

La conexiones entre los contenedores del biorreactor y el sistema neumático se logran empleando mangueras de silicona de 6 mm de

diámetro interno, capaces de soportar la presión y temperatura necesarias para su esterilización. Para comunicar a cada recipiente con el sistema neumático se utilizan acoples reductores. A la salida de 6 mm del conector se inserta una manguera de 5 cm de longitud la cual lleva a un filtro de membrana de 0,22  $\mu\text{m}$ , que esteriliza el aire proveniente del sistema neumático. Luego de ese filtro se conecta una última manguera que lleva a un conector simple en la tapa del recipiente.

Para comunicar el recipiente que contiene las plantas con el recipiente que contiene el medio de cultivo se utiliza una manguera de 30 cm de longitud, conectada a cada tapa mediante picos dobles.

### 2.1 Sistema Neumático

El proceso de insuflar aire a los biorreactores consta de tres etapas:

- (1) Filtrado y regulación de la presión de aire, ambos realizados mediante un filtro-regulador.
- (2) Direccionamiento del aire realizado mediante electroválvulas 3/2 neumáticas.
- (3) Distribución de aire mediante el sistema de mangueras neumáticas.

Debido a que el pasaje de medio de cultivo entre los dos recipientes de cada biorreactor debe ocurrir en ambos sentidos, es necesario insuflar aire en un recipiente al mismo tiempo que se libera al exterior el aire contenido en el otro. Para ello se instala un par de válvulas por cada piso del rack destinado a la propagación.

### 2.2 Sistema Lumínico

Se colocan arreglos de tiras de LEDs en cada piso ya que este tipo de iluminación posee una baja disipación de temperatura. En cada piso del sistema destinado a la propagación, se fijan nueve tiras de LED de 70 cm de longitud en la pared posterior, espaciando las mismas cada 1 cm focalizando así la iluminación en los biorreactores. La iluminación entre los pisos alterna en LEDs blancos, rojos y azules para lograr distintas condiciones de propagación. Estudios previos demuestran que la calidad de la luz tiene un fuerte impacto en el desarrollo de las plantas, siendo la luz roja y azul (y la combinación de las mismas), la que mayores efectos morfológicos tiene sobre las plantas (Kozai *et al.*, 2000).

### 2.3 Sensores

En lo que respecta a la medición de variables se requieren sensores de temperatura y humedad relativa de cada piso. Para esto se dispone de dos sensores *DHT-11*, ubicados uno en cada piso. Estos dispositivos poseen un sensor de humedad resistivo y un componente medidor de temperatura conectados a un microcontrolador de 8 bits.

### 2.4 Microcontrolador

El microcontrolador empleado es un *Arduino MEGA 2560* que al igual que el resto de las herramientas elegidas, es una plataforma de Software Libre y Hardware Abierto. Constructivamente la plataforma consta de una plaqueta que posee un microcontrolador *Atmel ATmega2560* que se conecta a 54 pines digitales de entrada/salida (de los cuales 15 pueden ser utilizados como salidas PWM), 16 entradas analógicas, 4 UARTs (*Universal Asynchronous Receiver-Transmitter*), un cristal oscilador de 16 MHz y una conexión USB

## 3. DESARROLLO DE SOFTWARE

Para lograr un sistema de supervisión disponible desde cualquier terminal se opta por el desarrollo de un Servidor Web mediante el cual se pueda acceder de manera remota a la totalidad de las funciones del Sistema de Inmersión Temporal.

- **Ubuntu versión 14.04:** Sistema operativo del servidor basado en GNU/Linux.
- **Apache versión 2.4.7:** Servidor web HTTP.
- **MySQL versión 5.5.47:** Sistema de gestión de bases de datos relacional.
- **Python versión 2.6:** Lenguaje de programación interpretado, multiplataforma y multiparadigma.
- **Django versión 1.8:** *Framework* desarrollado para el lenguaje *Python* cuyo objetivo principal es proveer herramientas para el diseño de Páginas Web y el manejo de bases de datos.

Complementando las funcionalidades de *Django* se emplean los lenguajes *JavaScript* y *AJAX*.

### 3.1 Implementación del patrón MVC/MTV en Django

El patrón MVC (*Model View Controller*) define una forma de desarrollar *Software* en la que el código para definir y acceder a los datos (el modelo) está separado del pedido lógico de asignación de ruta (el controlador), que a su vez está separado de la interfaz del usuario (la vista). Una ventaja clave de este enfoque es que los componentes tienen un *acoplamiento débil* entre sí. La interfaz gráfica consta de una Página Web que presenta los datos a partir del servidor que posee el sistema. El proyecto de *Django* denominado *bioweb* se encarga de gestionar la Página Web en la cual se muestran los datos y configuraciones posibles para el sistema ejecutándose sobre el servidor web *Apache*.

Dicho proyecto se conforma de siete aplicaciones, cada una encargada de resolver diferentes tareas del sistema de gestión del biorreactor. A continuación se describe brevemente cada aplicación.

*Alarmas* Gestiona el sistema de alarmas dentro del proyecto *bioweb*. Tiene incidencia en la base de datos, en la cual se alojan las alarmas activadas en algún momento del proceso y su estado actual. También se encarga de gestionar la confirmación de las alarmas por parte del usuario y las parametrizaciones de los niveles de disparo

*Contacto* La aplicación contacto se encarga de gestionar el envío de mensajes del usuario registrado directamente desde la Página Web a la dirección de correo de los desarrolladores del sistema.

*Hmi* Posee la tarea de realizar la visualización del esquemático del SIT conformado por los biorreactores y luces funcionando y sus parámetros dependiendo (o no) de la receta en curso.

*Log* Se encarga de gestionar los usuarios y permisos. Mediante el empleo de las aplicaciones de *Django* incluidas en la instalación base, se generan las opciones de crear, modificar y eliminar perfiles de usuarios, además de permitir gestionar diferentes jerarquías de permisos por medio de grupos. Se generan tres grupos con diferentes permisos: *Espectador*, sólo posee los permisos para ver el SIT, las alarmas y los mensajes, sin poder realizar modificaciones en ningún área del sistema; *Técnico*, posee los permisos de *espectador* y además la capacidad de confirmar alarmas, ejecutar órdenes y lanzar, cancelar, crear, eliminar y modificar recetas y *Administrador* posee los permisos de *técnico* mas la capacidad de generar y eliminar nuevos usuarios y modificar su grupo de pertenencia.

*Mensajes* Con el fin de notificar al usuario sobre los acontecimientos ocurridos en el sistema, se emplea esta aplicación para gestionar los mensajes en pantalla y de esta manera complementar el sistema de alarmas. Entre las notificaciones que realiza al usuario se incluyen cambios de estado del sistema, creación de nuevos usuarios, creación de recetas, lanzamiento de recetas en los distintos pisos, lanzamiento de órdenes y toma de medición de humedad o temperatura..

*Recetas* Con el fin de simplificar la utilización del sistema y administrar su funcionamiento, se emplea la aplicación *Recetas*. Ésta se encarga de ofrecer vistas para cargar nuevas recetas, editarlas o eliminarlas, guardándolas en la base de datos. Una vez cargadas, es posible asignarlas a un piso y gestionar su funcionamiento. En el caso que esté asignada una receta a algún piso, es posible ejecutar órdenes independientes en dicho piso instantáneamente mediante un sistema de selección.

### 3.2 Base de Datos y Tablas

Luego de definir el proyecto de *Django* y sus aplicaciones, el *framework* automáticamente instancia en la base de datos tablas homónimas a los

modelos definidos en el mismo. Es necesario tener en cuenta que debido a la técnica de ORM (*Object-Relational Mapping* o Mapeo Objeto-Relacional) los objetos del modelo que sean instanciados y modificados son convertidos por *Django* a filas dentro de la tabla a la cual dicho modelo esté relacionada, convirtiéndose de esta manera en objetos *persistentes*.

### 3.3 Controlador entre el Modelo y la Vista Física y Gestores

Habiendo definido la vista como la interfaz del usuario, se puede enmarcar al SIT como una de las dos vistas disponibles, siendo la otra la Página Web. En consecuencia es necesario el diseño de un controlador capaz de comunicar la vista física con los modelos de estado definidos en las aplicaciones de *Django*. Para ello se recurre a la implementación de objetos definidos como *Gestores* capaces de realizar la comunicación serial con el microcontrolador para el manejo de la vista y la actualización de los modelos almacenados en la base de datos. Los gestores son un conjunto de objetos definidos con el fin de tomar decisiones del lado del servidor con respecto al manejo lógico del SIT y la comunicación entre la base de datos y el microcontrolador. Para la implementación se hace uso de la librería *PySerial* de *Python* dedicada a la comunicación serie entre el servidor y el microcontrolador.

### 3.4 Comunicación Biorreactor - Servidor

Se describen a continuación, los objetos utilizados para implementar la comunicación entre el SIT y el Servidor Web.

*Mensajero* Es usado por los *Gestores* que precisan de una comunicación con el Arduino tanto para ejecutar órdenes como para realizar el sensado de variables. Al recibir un mensaje para enviar, el *Mensajero* lo codifica de manera tal que pueda ser interpretado por el Arduino y realiza su envío.

*Traductor* El *Traductor* es el objeto que se encarga de tomar la orden que recibe el *Mensajero* y codificarla de manera que sea comprendida por el microcontrolador. El mismo consta de un diccionario que contiene las posibles órdenes a mandar y su respectivo código.

*Comunicador* Una vez que el *Mensajero* recibe la orden codificada por parte del *Traductor*, hace uso del *Comunicador* para efectuar su envío.

### 3.5 Controlador del Sistema de Inmersión Temporal

La programación del *Arduino Mega2560* es realizada en su lenguaje nativo agregando el uso de las librerías *DHT11* *Timermaster* y *QueueArray*. Esta última se utiliza para la ejecución de las tareas de manera FIFO (*First Input First Output*). Todas las librerías son de carácter libre

Al recibir un mensaje vía puerto serie, el controlador clasifica el mensaje entre las órdenes y peticiones que tiene definidos. El controlador está atado a las órdenes del servidor y es incapaz de efectuar acciones de manera autónoma, relegando la lógica a la programación de alto nivel en el ordenador.

### 3.6 Interfaz Gráfica

La Interfaz Gráfica es desarrollada a través de plantillas *HTML* con el agregado del sistema de *etiquetas* de *Django* para la lógica de presentación y *JavaScript* para las operaciones *client-side*. Las librerías utilizadas son *jQuery*, *Highcharts* y *Bootstrap*. Una vez iniciada la sesión se puede acceder a las distintas partes de la interfaz gráfica. En la parte superior de la página se puede observar un *alarmero*, donde se visualizan las alarmas activas o no confirmadas; y un *mensajero*, donde se pueden ver los últimos cuatro mensajes reportados por el sistema. Debajo de estos se puede apreciar una barra de navegación con vínculos a las distintas funciones de la páginas:

- **SIT:** Refleja el estado actual del Sistema de Inmersión Temporal.
- **Recetas:** Recetas cargadas por los usuarios del sistema.
- **Graficos:** Gráficas de las variables muestreadas por los sensores.
- **Alarmas:** Totalidad de las alarmas, tanto las actuales como las previamente disparadas.
- **Mensajes:** Totalidad de los mensajes reportados por el sistema desde su comienzo.
- **Configuración:** Configuración de los niveles de alarmas y los tiempos de muestreo de las variables.
- **Usuarios:** Totalidad de los usuarios registrados en el sistema.
- **Contacto:** Ayuda y contacto con los desarrolladores.

*Pantalla SIT* Al presionar el primer vínculo, titulado *SIT*, el usuario accede a una pantalla similar a la vista en la Figura 3. Esta página



Fig. 3. Página de Interacción con el Sistema de Inmersión Temporal

está dividida en dos partes: un menú por cada piso del SIT indicando el estado de la receta

en el piso en caso de haberle asignado una y un modelo representativo del estado actual del SIT. En el menú se pueden apreciar dos cuadros indicando el piso al que corresponde y el estado. En caso de no haber asignado receta alguna se muestra un texto indicando que el piso está libre de recetas y dos desplegables: el primero permite operar al SIT de manera manual y el segundo desplegable permite seleccionar una receta de la lista de recetas disponibles, mostrando su nombre y la especie de planta a la que corresponde. Debajo de la información de la receta se encuentra un grupo de botones (*Ejecución*, *Parada* y *Pausa*).

*Pantalla Recetas* En esta página el usuario puede apreciar la totalidad de las recetas cargadas. Las características más relevantes de configuración de una receta son:

- **Especie:** Especie de la planta a la que corresponde la receta.
- **Duración de la receta:** Tiempo de duración de la receta expresado en días.
- **Tiempo de Inmersión:** Tiempo de duración de la alimentación de las plantas expresado en minutos.
- **Duración del día:** Tiempo de duración del ciclo diurno (iluminación activa) expresado en horas.
- **Duración de la noche:** Tiempo de duración del ciclo nocturno (iluminación inactiva) expresado en horas
- **Cantidad de inmersiones:** Cantidad de inmersiones que recibirán las plantas cada 24 horas.

*Pantalla Gráficos* Contiene los gráficos de la evolución temporal de las variables tomadas por los sensores del biorreactor con disponibilidad de exportación en múltiples formatos.

*Pantalla Alarmas* En esta página se presenta la totalidad de las alarmas activadas desde el inicio del sistema en forma de tabla. Se pueden apreciar los datos característicos de cada una. Las alarmas se presentan en cuatro colores diferentes:

- **Rojo:** Alarmas que no hayan sido ni confirmadas ni corregidas.
- **Amarillo:** Alarmas que fueron confirmadas pero no corregidas.
- **Celeste:** Alarmas que fueron corregidas pero no confirmadas por el operador.
- **Verde:** Alarmas que fueron tanto confirmadas como corregidas.

*Pantalla Mensajes* En esta página están listados todas las acciones reportados por la página con la hora en la que estas fueron realizadas.

### 3.7 Pantalla Configuración

Brinda al usuario la posibilidad de modificar los niveles de alarma alto y bajo para cada variable

en cada piso disponible. Además puede modificar los tiempos de muestreo de cada variable.

### 3.8 Pantalla Administración de Usuarios

Contiene la información de todos los miembros de la página, la cual indica (entre otros) nombre de usuario, ID, correo electrónico y grupo al que pertenece.

### 3.9 Pantalla Contacto

Contiene información acerca de los desarrolladores y permite descargar un manual de operación.

## 4. RESULTADOS

El experimento planificado contempla el lanzamiento de una receta de 30 minutos de duración. Durante dicho tiempo se somete al Sistema a cambios generados por usuarios pertenecientes a diferentes grupos y se fuerzan cambios en las condiciones del Sistema con el fin de evaluar su respuesta. La receta diseñada es lanzada a las 16:04:37 por el usuario Administrador. En el transcurso del experimento se forzaron dos alarmas por temperatura alta en el piso superior a las 16:36:17 y 16:21:44 quedando las mismas registradas en el alarmero y en los correos electrónicos de los usuarios. La receta finaliza a las 16:34:38 dejando en funcionamiento únicamente el control dedicado a la iluminación como se demuestra en el mensaje reflejado. A las 16:35:37 el usuario Técnico detiene la receta. A las 16:45:52 se desconecta el microcontrolador del servidor generando una falla de comunicación. A las 16:46:20 se reconecta el microcontrolador al servidor restableciendo la continuidad del proceso.

A modo de ilustración, la evolución de las variables de temperatura se pueden apreciar en la Fig. 4.

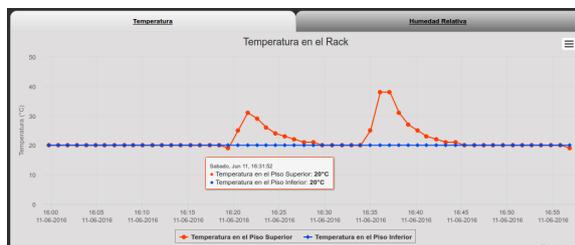


Fig. 4. Valores de temperatura registrados durante el experimento.

## 5. CONCLUSIONES

En este trabajo se desarrolla un Sistema de Inmersión Temporal automatizado a medida para un laboratorio dentro de la Universidad Nacional de Quilmes. Gracias a los materiales seleccionados para su construcción, el sistema posee un costo considerablemente inferior al de los sistemas comerciales que además cuentan con menor cantidad

de prestaciones al construido. Es implementado con las opciones de manejo manual, la cual mantiene las funciones que los usuarios conocen, y manejo automático, permitiendo a los usuarios cargar recetas personalizadas, hacer un seguimiento y analizar los resultados obtenidos. Debido a una implementación modular y de acoplamiento débil del *Software* empleando el patrón de diseño MVC y un amplio repositorio provisto por herramientas libres, se pueden generar fácilmente cambios en el programa enfocados a sectores específicos sin necesidad de cambios significativos. El sistema de adquisición de datos permite tener un registro de todas las eventualidades ocurridas durante el proceso dándole al mismo una herramienta capaz de registrar cualquier cambio en el estado del sistema y permitiéndole analizar todos los factores que intervienen en el proceso de propagación.

## REFERENCIAS

- Escalona, M, JC Lorenzo, B González, M Daquinta, JL González, Y Desjardins and CG Borroto (1999). Pineapple (ananas comosus l. merr) micropropagation in temporary immersion systems. *Plant Cell Reports* **18**(9), 743–748.
- Etienne, H and M Berthouly (2002). Temporary immersion systems in plant micropropagation. *Plant Cell, Tissue and Organ Culture* **69**(3), 215–231.
- Georgiev, Vasil, Anika Schumann, Atanas Pavlov and Thomas Bley (2014). Temporary immersion systems in plant biotechnology. *Engineering in Life Sciences* **14**(6), 607–621.
- Imanishi, Leandro, Alice Vayssières, Claudine Franche, Didier Bogusz, Luis Wall and Sergio Svistoonoff (2011). Transformed hairy roots of discaria trinervis: a valuable tool for studying actinorhizal symbiosis in the context of intercellular infection. *Molecular Plant-Microbe Interactions* **24**(11), 1317–1324.
- Kozai, T, C Kubota, SMA Zobayed, QT Nguyen, F Afreen-Zobayed and J Heo (2000). Photoautotrophic (sugarfree medium) micropropagation. In: *Proc. Workshop on contamination and acclimatization management in plant cell and tissue culture*. pp. 5–19.
- Nations, United (1992). Convention on biological diversity. *Use of terms. Artículo 2*.
- System, National Communications (2004). Supervisory control and data acquisition (scada) systems. Technical report. Office of the Manager National Communications System.
- Werner, Gijsbert DA, William K Cornwell, Janet I Sprent, Jens Kattge and E Toby Kiers (2014). A single evolutionary innovation drives the deep evolution of symbiotic n2-fixation in angiosperms. *Nature communications*.