

Proyecto Hevelius

Empresa DevNull

Requerimientos

Carlos Guajardo Miranda

Jefe de Proyecto

cguajard@alumnos.inf.utfsm.cl

cel. 09-95046118

Marina Pilar Daza

Miembro del Equipo

mpilar@alumnos.inf.utfsm.cl

cel. 09-84085407

Esteban Espinoza Martínez

Miembro del Equipo

eespinoz@alumnos.inf.utfsm.cl

cel. 09-85596939

Tomás Staig Fernández

Miembro del Equipo

tstaig@alumnos.inf.utfsm.cl

cel. 09-97615666

13 de abril de 2007

1. Resumen Proyecto

El proyecto Hevelius, a cargo de la pre-Empresa DevNull, pretende, como objetivo principal, crear un software de control de telescopios, que sirva de impulso hacia la creación de un control genérico de telescopios. Hevelius, debe poseer flexibilidad en su estructura como software, de manera que permita la futura implementación de funcionalidades extras y el reemplazo de otras.

“Control de telescopio amateur bajo interfaz de control profesional”

2. Lista de Requerimientos

2.1. Requerimientos Funcionales Propias de la Aplicación

- Ajustar posición del telescopio bajo sistema de coordenadas ecuatoriales.
- Mover el telescopio a la velocidad de la hora sideral.
- Impedir observación a lugares donde exista Luminosidad Lunar.
- Mostrar un modelo visual del estado físico del telescopio.
- Ajustar manualmente el telescopio.
- Detener de forma inmediata el telescopio en caso de emergencia.

2.2. Requerimientos Funcionales Genéricos

- Guardar coordenadas de observaciones realizadas.
- Controlar acceso a la aplicación (sesiones).

2.3. Requerimientos Funcionales Distintivos

- Crear software bajo plataforma ACS.

2.4. Requerimientos No Funcionales Distintivos

- Permitir conexión a sistema diferente de Pointing Automático.
- Documentar, crear manuales y/o informes técnicos con formato estándar de ALMA.

3. Especificaciones del Software

La plataforma en la que se desarrollará el proyecto se basa en los requerimientos de nuestro cliente, la cual consiste en el Sistema Operativo Linux y el software ACS 6.0.

3.1. Especificación de Hardware

Los equipos con los cuales se desarrollará nuestro proyecto poseen de las siguientes características:

- Procesador Pentium IV 3.0 Ghz de 32 bits.
- 1Gb RAM
- Telescopio NEXSTAR 4 SE que posee computador manual de 16 caracteres con pantalla LCD y 19 botones. Algunas de sus características: apertura de 102 mm, largo focal de 1325 mm, radio focal de 12.99 mm, entre otras cosas.
- Se añadirá al telescopio un CCS para poder obtener imágenes.

El riesgo que se corre con respecto a la adquisición del hardware es la obtención del telescopio, puesto que tanto el resto del equipo a utilizar y la plataforma de software ya están disponibles. Sin embargo, a la fecha de la creación de este informe, se informó que el Telescopio será adquirido dentro de esta semana (9 al 13 de abril), por lo que el riesgo de hardware se reduce considerablemente.

4. Presentación Cliente y Usuario

Hevelius se enmarca en un proyecto de mayor magnitud, llamado “Software development for ALMA-CONICYT: Building up expertise to meet ALMA-CONICYT software requirements within a Chilean University”. Los representantes de dicho proyecto son Mauricio Araya y Horst Von Brand.

Mauricio Alejandro Araya López, estudiante de Ingeniería Civil Informática y Magíster en Informática en la Universidad Técnica Federico Santa María. Se ha desempeñado como Team Leader y Partnership del ACS-UTFSM, donde colaboró en el ALMA Common Software, liderado por European Southern Observatory en diversas aplicaciones sobre ACS y en el desarrollo del mismo framework ACS.

Horst von Brand, PhD en Computer Science de la Louisiana State University. Actualmente se desempeña como profesor del Departamento de Informática de la Universidad Técnica Federico Santa María, además de ser Jefe del Proyecto “Software development for ALMA-CONICYT: Building up expertise to meet ALMA-CONICYT software requirements within a Chilean University”.

4.1. Contacto

A través de la página web del proyecto www.acs.inf.utfsm.cl A Mauricio Araya, en el Laboratorio de Sistemas Distribuidos, Departamento de Informática, Edificio F2, Piso 1, UTFSM; o mediante correo electrónico maray@inf.utfsm.cl. A Horst von Brand, a través de la Secretaría del Departamento de Informática, Edificio F2, Piso 1, UTFSM; o mediante su correo electrónico vonbrand@inf.utfsm.cl.

5. Anexo: Detalle de los Requerimientos

5.1. Requerimientos Funcionales Propias de la Aplicación

5.1.1. Ajustar posición del telescopio bajo sistema de coordenadas ecuatoriales.

Es necesario ajustar el telescopio a una posición fija, esto debe hacerse respecto al sistema de coordenadas ecuatoriales R.A.DEC., el cual permite conocer la ubicación de la estrella por periodos, es decir, se puede considerar que la estrella no se mueve en periodos cortos de tiempo. Pero todavía queda el problema de que la tierra se mueve a medida que pasa el tiempo. Para esto, se considera la hora sideral, la cual considera estos movimientos, de manera que a una misma hora en días distintos, durante alguno de estos periodos se puede ver la estrella en el mismo lugar. El presetting no puede demorar mucho, por lo que los movimientos del telescopio tienen que ser relativamente rápidos ¹.

5.1.2. Mover el telescopio a la velocidad de la hora sideral.

El sistema debe dar la posibilidad de tracking, de manera que el telescopio pueda seguir el punto que se está observando a través del tiempo, a pesar de los movimientos de la tierra. Para esto, se debe implementar un sistema distinto al presetting, en el sentido que este tiene que ser más sutil, ya que los movimientos son mucho más pequeños, y no queremos que el telescopio se mueva mucho para no afectar a los estudios que se puedan estar haciendo.

5.1.3. Impedir observación a lugares donde exista Luminosidad Lunar.

Al observar las estrellas en presencia de la luna, uno se percata que existe una disminución en la calidad de la observación debido a la luminosidad de la luna y a su volumen. El software Hevelius debe advertir al controlador del telescopio que este se dirige a un punto en donde la observación no será de calidad, por lo que se cuidarán tanto el tiempo de observación como los lentes de los telescopios sensibles a mucha luminosidad.

5.1.4. Mostrar un modelo visual del estado físico del telescopio.

Un operador de telescopio puede estar situado: en el telescopio mismo, en un centro de control, o en cualquier otro lugar, por lo que no siempre tendrá la oportunidad de observar cuál es el movimiento real del telescopio que controla. Hevelius debe simular el movimiento del telescopio para que el operador, esté donde esté, pueda tener una visión del comportamiento de este y así permitir diagnosticar alguna falla en la observación.

¹Más Información: http://es.wikipedia.org/wiki/Coordenadas_ecuatoriales

5.1.5. Ajustar manualmente la posición del telescopio.

Al realizar las tareas de presetting y tracking existe la posibilidad de que no se vea lo que se supone, es decir, que exista un ligero corrimiento de lo que se debiera ver, por diversos motivos, como son las características de material del telescopio, ya que en algunas posiciones este puede doblarse por acción de su propio peso, generando una pequeña diferencia con lo esperado. Es por esto que se requiere de un sistema de pointing manual, que permita al operador del telescopio realizar movimientos manuales para ajustar el telescopio. Cabe destacar, que este sistema de pointing debe ir por debajo del presetting y del tracking, afectando a lo que se ve realmente, pero sin cambiar las coordenadas y horas con las que ellos trabajan.

5.1.6. Detener de forma inmediata el telescopio en caso de emergencia.

En general, los telescopios son aparatos delicados a condiciones climáticas, de luz y muchos otros factores. Es por esto que se debe contar con una opción para proteger al telescopio de estas situaciones. Esta opción es un botón de pánico que debe hacer que el telescopio quede protegido de manera “instantánea” y, debe lograrlo, considerando que trabaja sobre la plataforma ACS de manera distribuida, es decir, puede estar siendo operado de cualquier parte del mundo. Se debe asegurar que aún en estas situaciones, el botón de pánico funcione rápidamente.

5.2. Requerimientos Funcionales Genéricos

5.2.1. Guardar coordenadas de observaciones realizadas.

Cada vez que se utiliza Hevelius se realiza una serie de investigaciones u observaciones que son de gran importancia para los que lo realizan, es por ello que Hevelius guardará un registro de todas las coordenadas de observación que se realicen durante una sesión, generando un reporte con la información obtenida.

5.2.2. Controlar acceso a la aplicación (sesiones).

Para tener un control de la gente que usará Hevelius, se implementará un sistema de sesiones, en donde cada usuario debidamente registrado, ingresará al software para realizar sus trabajos.

5.3. Requerimientos Funcionales Distintivos

5.3.1. Crear software bajo plataforma ACS.

La plataforma ACS (Alma Common Software) es un framework que facilita el uso de sistemas distribuidos. Se caracteriza por su modelo Componente-Container, el cual permite utilizar aplicaciones desarrolladas en

otro sitio como si estuvieran donde uno desarrolla. Hevelius debe utilizar aplicaciones ya creadas (para no replicar código), crear aplicaciones para el resto del mundo y utilizar todo el sistema en sí para contruir el software.

5.4. Requerimientos No Funcionales Distintivos

5.4.1. Permitir conexión a sistema diferente de Pointing Automático.

El pointing, posicionamiento del telescopio en alguna coordenada, puede ser desarrollado de diversas formas, según el telescopio a utilizar. Hevelius será capaz de poder ajustarse a nuevos sistemas de pointing automático, lo cual permite a Hevelius ser un software modular.

5.4.2. Documentar, crear manuales e/o informes técnicos con formato estándar de ALMA.

El proyecto Hevelius al enmarcarse en el proyecto de ALMA y Conicyt “Software development for ALMA-CONICYT: Building up expertise to meet ALMA-CONICYT software requirements within a Chilean University”, debe cumplir ciertos estándares, como son el formato y lenguajes de programación. Toda la documentación del proyecto y comentarios en el código fuente estarán realizados en Inglés, puesto que el proyecto tiene un carácter internacional. Otro estándar a seguir, son los tipos de lenguajes permitidos, puesto que el proyecto funciona con el framework ACS 6.0, Hevelius puede utilizar sólo los lenguajes Java, Python, C y C++.