

Facultad de Informática.
Ingeniería en Informática.
Ingeniería del Software.
Proyecto: Everywhere House Control.

Gestion y planificacion

Creado por	Gutierrez, Hector	Guzman, Fernando
	Ladrón, Alejandro	Maldonado, Miguel Alexander
	Morales, Álvaro	Ochoa, Victor
	Rey, José Antonio	Saavendra, Luis Antonio
	Tirado, Colin	Vicente, Victor

Índice general

1. Introducción	1
1.1. Propósito	1
1.2. Ámbito	1
1.3. Referencias	1
1.4. Perspectiva general	2
2. Visión general	3
2.1. Propósito, alcance y objetivos del proyecto	3
2.2. Suposiciones y restricciones	3
2.3. Productos del proyecto	3
3. Organización	5
3.1. Estructura organizativa	5
3.2. Modelo de proceso	5
3.3. Contactos externos	6
3.4. Roles y responsabilidades	6
3.4.1. Hardware	6
3.4.2. Software	6
3.4.3. Servidor	7
4. Proceso de gestión	8
4.1. Estimaciones del proyecto	8
4.2. Plan de proyecto	9
4.2.1. Módulos	9
4.2.2. Recursos del proyecto	10
4.2.3. Entregas	10
4.2.4. EDT	10
4.3. Planificación del proyecto	11
4.4. Sprint	11
4.5. Control y gestión del proyecto	12
4.5.1. Plan de gestión de requisitos	12
4.5.2. Plan de control de la planificación	12
4.5.3. Plan de control de calidad	12
4.6. Plan de gestión de riesgos	12

5. Plan de procesos técnicos	13
5.1. Esquema de desarrollo	13
5.2. Métodos, herramientas y técnicas	13
6. Planes de soporte del proceso	14
6.1. Plan de evaluación	14
6.2. Plan de documentación	14

Capítulo 1

Introducción

El plan de desarrollo del proyecto EHC, está caracterizado por su alto contenido hardware, por lo tanto, el modelo de desarrollo, la organización y gestión, y la planificación, tienen distintas peculiaridades, que en ocasiones difieren bastante del modelo de proceso software. Como este proyecto está enmarcado dentro de la asignatura de Ingeniería del Software, se ha buscado siempre, establecer las bases fundamentales lo más semejantes posibles al desarrollo software. Destacar que en el desarrollo de prototipos hardware, muchos de ellos se acaban reutilizando lo que aumenta la productividad por rendimiento asociada.

1.1. Propósito

El propósito de este documento es describir el plan de gestión del proyecto, que se pretende seguir durante todo el curso académico de la asignatura. Esta dirigido, tanto a los participantes en el proyecto, como al profesor de la asignatura, que también forma parte importante del proyecto, al ser el que coordinara las entregas actuando como auditor del proyecto. Se presenta a los participantes y se detalla la organización que existe entre ellos.

1.2. Ámbito

Este plan de proyecto se enmarca dentro del ámbito universitario, en vista a un año y ligado a la enseñanza del desarrollo software. Formando parte de un proyecto de la facultad de Ingeniería Informática de la Universidad Complutense de Madrid, que se desarrollara durante todo el curso académico 2013/2014.

1.3. Referencias

- Documento de Visión.
- Documento de Requisitos.
- Documento de arquitectura del software.
- Documento de plan de pruebas.

- Documento de plan de gestión de riesgos.

1.4. Perspectiva general

El plan de desarrollo de software comprende la siguiente información:

- Resumen del proyecto: contiene una descripción del propósito, el alcance y los objetivos del proyecto. También define los elementos que se han de generar durante el desarrollo del proyecto.
- Organización del proyecto: describe la estructura organizativa del equipo de desarrollo.
- Proceso de gestión: detalla los costes estimados y la planificación, define las fases e hitos del proyecto, y describe el seguimiento del proyecto.
- Planes del proceso técnico: proporciona una visión general del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas a utilizar.
- Planes del proceso de soporte: contiene el plan de gestión de la configuración.

Capítulo 2

Visión general

2.1. Propósito, alcance y objetivos del proyecto

El producto tiene como objetivo, facilitar el uso y la adaptación a los usuarios inexpertos, mediante un acceso rápido y directo, de los dispositivos del hogar. Para una explicación más detallada, revisar el documento de visión.

2.2. Suposiciones y restricciones

Los participantes en el proyecto son en su totalidad estudiantes universitarios. El proyecto carece de presupuesto, más allá de las inversiones a título personal de los integrantes del mismo. El grupo está compuesto por 10 personas, con poca experiencia en trabajo en grupo. La fecha de finalización del proyecto coincide con la del curso académico 2013/2014.

Los conocimientos requeridos son:

- Programación dispositivos hardware Arduino y Raspberry PI.
- Programación IOS y Android.
- Lenguajes de programación de base de datos.
- Comunicaciones inalámbricas.
- Domótica del hogar, funcionamiento específico.

2.3. Productos del proyecto

Se generarán los siguientes productos durante el desarrollo del proyecto:

- Plan de desarrollo de software: planificación en la que se pretende apoyar para la elección de las tareas asociadas al sprint.
- Plan de gestión de riesgos: documento donde se detallan los planes de prevención, mitigación o contingencia para solventar los posibles riesgos durante el proceso de desarrollo del proyecto.

- Especificación de Requisitos: formulación de los requisitos, tanto funcionales como no funcionales, que definirá los casos de uso del sistema.
- Glosario: definición de términos usados en los documentos del proyecto.
- Visión: análisis y definición de las principales características y restricciones del sistema, haciendo hincapié en la funcionalidad requerida por los usuarios finales.
- Plan de pruebas: documento que explica el proceso de gestión de calidad y comprobaciones sobre el software producido.
- Documento de la Arquitectura: detalla aspectos relativos al diseño y estructura del hardware y del software que intervienen en el proyecto.
- Casos de uso: descripción de los distintos usos del sistema. Dominado por la funcionalidad que se ofrece al usuario, definidas éstas en los requisitos.
- Planes de iteración: detallan el trabajo y planificación de cada iteración del proyecto. En nuestro caso dirigido a las mejoras incrementales de los prototipos.
- Entrega de prototipo ejecutable a finales de enero.
- Entrega de la versión final del proyecto en junio.

Capítulo 3

Organización

3.1. Estructura organizativa

La estructura del sistema que se está desarrollando tiene tres partes: parte de Hardware, parte de cliente o Software y parte de Servidor, pero será detallada más adelante. Debido a esta estructura, se ha escogido una organización de equipo de tipo “Descentralizado Controlado”. En esta organización se han definido tres departamentos (Hardware, Software y Servidor), que se encargan de cada uno de los tres pilares del sistema. En cada departamento hay un “Gestor de Departamento” que se encarga de que la evolución sea constante.

Además se ha definido también el rol de “Documentador”, que se encargará de organizar la documentación y de asegurarse de que sea lo más completa y útil posible. Las tareas de este rol se compaginan con las tareas propias de su departamento. Hay también un “Gestor Funcional del Sistema”, cuya función es asegurarse del buen funcionamiento de la comunicación entre los departamentos.

No obstante tanto el Gestor Funcional del Sistema, como los Gestores de Proyecto como los Documentadores son también desarrolladores, es decir, todo el personal dedicado al proyecto está involucrado activamente en las labores de desarrollo del sistema.

Por último, hay que hablar del cliente. Este proyecto no surge porque un tercero haya contactado con una empresa, para contratar un servicio o comprar un producto, sino que tiene un fin didáctico primordialmente. Teniendo esto en cuenta, el cliente será aquella persona/s que acoten, de alguna manera, los requisitos del sistema.

3.2. Modelo de proceso

El modelo de proceso que se sigue en el proyecto es uno basado en prototipos, combinado con una metodología ágil Scrum. La razón de escoger este modelo es la naturaleza del sistema que se está desarrollando. El núcleo del proyecto no tiene demasiada envergadura, sino que es en las numerosas funcionalidades donde recae la verdadera complejidad. Como la mayoría de éstas son independientes unas de otras, se pueden desarrollar por separado. Esto permite diseñar “sprints” de Scrum, de tal forma que, al acabar cada sprint (o incluso a mitad de él) se

tenga un sistema totalmente funcional (reducido respecto al producto final) que es susceptible de ser modificado más adelante, total o parcialmente, mediante alguna otra tarea que mejore o modifique su comportamiento, en otras palabras, se tenga un prototipo.

Para organizar el modelo de proceso, se ha hecho coincidir el contenido de los sprints de Scrum con los prototipos que resultan de cada iteración.

El número de reuniones semanales se reduce a dos veces por semana por norma general. Esto nos ofrece más agilidad a la hora de desarrollar el sistema ya que se desarrolla en el comienzo en tres módulos bien diferenciados (Software, Hardware y el servidor) y que no necesitan (en ese momento) integrarse en un único sistema. A medida que se va desarrollando cada módulo este número de reuniones irá aumentando ya que se necesita una mayor comunicación para integrar los distintos módulos en un único sistema.

El Scrum Master del proyecto es Álvaro Morales

3.3. Contactos externos

Stakeholders fuera del personal del proyecto:

- Federico Peinado (Cliente)
- Samer Hassan (Cliente)

3.4. Roles y responsabilidades

3.4.1. Hardware

- Héctor Gutiérrez (Gestor de Departamento; Desarrollador)
- Álvaro Morales (Gestor Funcional del Sistema; Desarrollador)
- Víctor Ochoa (Documentador; Desarrollador)
- Fernando A. Guzmán (Desarrollador)

3.4.2. Software

- Víctor (Gestor de Departamento; Desarrollador)
- Miguel A. Maldonado (Desarrollador)
- Colin Tirado (Documentador; Desarrollador)

3.4.3. Servidor

- Alejandro Ladrón (Gestor de Departamento; Desarrollador)
- Jose A. Rey (Desarrollador)
- Luis A. Saavedra (Desarrollador)

Capítulo 4

Proceso de gestión

4.1. Estimaciones del proyecto

Los recursos financieros de los que se disponen son limitados. Serán los participantes los que tendrán que invertir su dinero en el proyecto. Actualmente, se ha establecido un límite de 200 euros.

Como no poseemos datos estadísticos sobre los participantes del proyecto, se considerara la productividad de todos los participantes por igual. Así, la productividad del grupo será de 10 pm (personas mes). Considerando por lo tanto que $1 \text{ pm} = 20 \text{ pd}$ (personas día).

La estimación del coste del proyecto, se ha realizado conforme a horas de esfuerzo por persona. Dando una productividad al alumno de un 67 %. Si el curso universitario consta de 270 días, entonces un estudiante puede ser capaz de aprovechar 180 días. Con un rendimiento medio al día de 1 hora, una persona tendrá una productividad de 180 horas. De esta forma, al ser 10 estudiantes universitarios se estima que costará 1800 horas. Para la planificación no se puede contar con el total de horas, puesto que no existe proyecto sin contratiempos. Por esta razón se ha considerado la regla del 80 %. Ésta dice, “para poder obtener un coste real aproximado al coste esperado debemos planificar el 80 % del tiempo, puesto que el tiempo restante se invierte en imprevistos”.

En este proyecto tendremos dos peculiaridades añadidas que nos obligan a volver a aplicar dicha regla. El modelo de proceso usado (SCRUM) y el tipo de estrategia de desarrollo (desarrollo por prototipos). Esto lo tendremos en cuenta en la estimación, pero para que la planificación no se vuelva trípica se añadirán como tareas estimadas los porcentajes de incertidumbre. Estimándose los imprevistos debidos a estas peculiaridades, quedando el tiempo a planificar en el 80 % del coste total, es decir, en 1440 horas. Porque, aunque es cierto que tendremos imprevistos, los prototipos nos aportan la reutilización de los mismos, abaratando costes y aumentando la productividad por hora invertida.

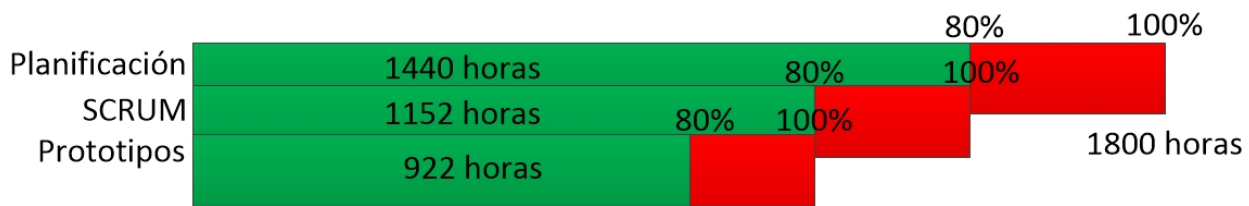


Figura 4.1: Estimación obtenida

4.2. Plan de proyecto

4.2.1. Módulos

Los módulos tienen las funcionalidades agrupadas por prioridad. De esta forma los sprint primero se centraran en las del módulo 1 y acabarán en el 3. Garantizando, que la aparición de algún riesgo afecte lo menos posible. También busca orientar la administración hacia el desarrollo software RUP, pretendiendo así el mejor seguimiento del proyecto por el profesor (cliente). Pudiendo asemejarse a funcionalidades por iteración.

- Módulo1
 - Control remoto de luces.
 - Control de dispositivos por infrarrojos.
 - Control remoto de la puerta del garaje.
 - Control remoto de persianas.
 - Aplicación IOS y Android.
- Módulo2
 - Control remoto del telefonillo.
 - Control de enchufe.
 - Previsión meteorológica.
 - Información de sensores meteorológicos.
 - Información de sensores de movimiento.
 - Aplicación web.
- Módulo3
 - Perfiles de mandatos.
 - Control remoto de Hardware.
 - Modo offline.
 - Control remoto de climatización.
 - Recolección de datos sobre el uso de dispositivos.
 - Monitorización consumo energético en tiempo real.

4.2.2. Recursos del proyecto

Los 10 componentes del grupo son:

Héctor Gutiérrez, Fernando Arturo Guzmán Almonte, Álvaro Morales Lozano, Luis Antonio Saavedra Palacios, Víctor Vicente, Miguel Alexander Maldonado Lenis, Alejandro Ladrón de Guevara, Víctor Ochoa Rodríguez, José Antonio Rey Álvarez, Colin Tirado Caamaño.

El calendario laboral asociado a cada uno de ellos es de 7 horas a la semana. Los recursos materiales de los que se dispone son:

- Arduinos 3
- Raspberry PI 1
- Protoboard 3
- Servidor 1
- Shield 2
- Conectores 2
- Maletines 1
- Entrenadores *(libres en horario de clase facultad informática UCM)
- Soldadores 2

Todos ellos con disponibilidad total de horario salvo los entrenadores.

4.2.3. Entregas

Este proyecto tiene 2 entregas importantes:

- 22 enero. Se entregara toda la documentación y una demostración de un prototipo parcial de una funcionalidad básica.
- Mayo. Sera la entrega final del proyecto, que llevara un prototipo alpha estable del proyecto.

4.2.4. EDT

La estructura de descomposición del trabajo sigue el diagrama de la figura [4.2](#)

Conforme a este, se han sacado las tareas de las funcionalidades definidas en los requisitos del proyecto.

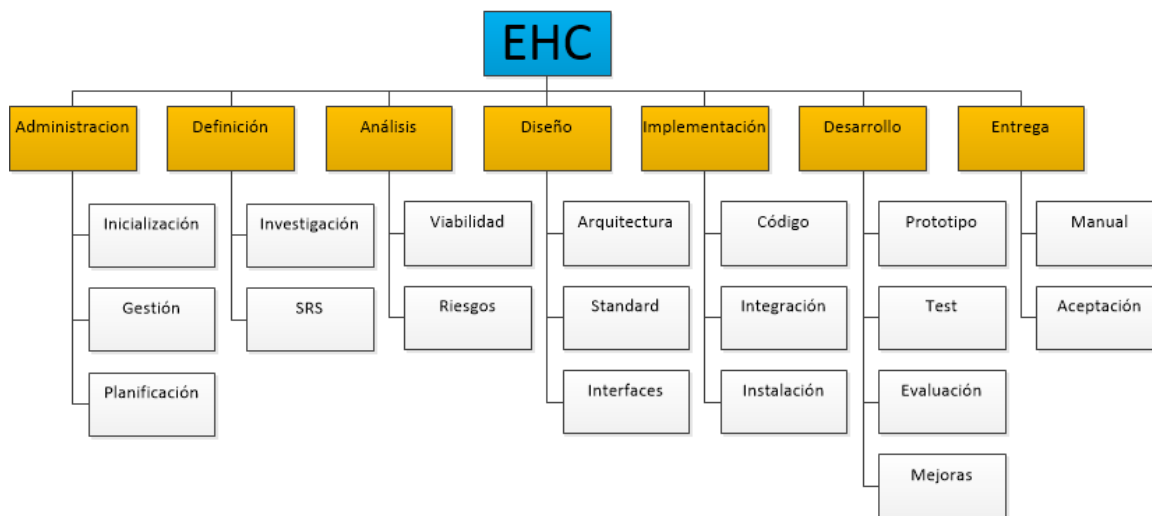


Figura 4.2: Estructura de descomposición del trabajo

4.3. Planificación del proyecto

La planificación del proyecto se ha llevado a cabo utilizando una herramienta de gestión de proyectos llamada MS Project 2010. En él, se tienen todas las tareas asociadas a las funcionalidades que se quieren implementar. Puesto que no puede meterse en este documento, se adjunta un archivo llamado “Microsoft Project - Plan de proyecto.21enero.pdf” con una captura de esta planificación.

Esa planificación, tiene desarrollada todas las tareas que se tendrán que hacer para ofrecer cierta funcionalidad. Pero éstas, se irán haciendo, conforme al sprint a los que pertenezcan. Dando nos la facilidad de elegir la tarea que más repercusión tenga (tarea crítica) o la que más funcionalidad nos ofrezca (cuello de botella).

4.4. Sprint

Se ha decidido tener 3 sprint en el primer cuatrimestre, dirigidos a poder ofrecer un prototipo con funcionalidad infrarroja, en vistas a la primera entrega que será en enero.

Para el segundo cuatrimestre se espera poder hacer sprint más cortos, ya que no se desarrollara tanta documentación.

4.5. Control y gestión del proyecto

4.5.1. Plan de gestión de requisitos

Se prevé que los requisitos puedan cambiar, pero será fundamentalmente por mejoras en la fase de desarrollo, ligadas a la funcionalidad de los prototipos. Por lo que, más bien forma parte del desarrollo iterativo incremental.

4.5.2. Plan de control de la planificación

Al tener desarrolladas las tareas y estimas, el control del proyecto se lleva por cantidad de trabajo realizado, es decir esfuerzo realizado por unidad de tiempo. En nuestro caso esta relación será de 1 a 1 por simplicidad. Así podremos seguir la progresión del proyecto en un diagrama burn down que nos ofrece la herramienta CASE (MS Project 2010).

4.5.3. Plan de control de calidad

Para garantizar al cliente el correcto funcionamiento se tienen los test, que nos garantizaran su fiabilidad. Pero para garantizar la calidad del proyecto se tienen previstas como tareas ya consideradas (lanzadas en la planificación), revisiones técnicas formales sobre los diseños y las arquitecturas (no aparecen con ese nombre por especificar más la tarea). Que son la parte más críticas en hardware. Aunque estas se llevaran a cabo tras algún test, sea o no satisfactorio, puesto que no es tan fácil depurar hardware.

Al formar parte de la planificación, se puede decir que esas revisiones forman parte del proyecto, ya que se han contabilizado. Ofreciendo así una garantía de calidad mínima.

4.6. Plan de gestión de riesgos

Gracias a la detallada descomposición y a que estas sean tareas asequibles. Los riesgos asociados a índices catastróficos han desaparecido. Pero es cierto que podrían aparecer riesgos que también tengan repercusión notable sobre la funcionalidad, inicial tal y como se detalla en el documento de riesgos. La periódica actualización de la planificación nos avisara de la posible aparición, ofreciéndonos la posibilidad de tratar la mayor parte de ellos con anterioridad.

Capítulo 5

Plan de procesos técnicos

5.1. Esquema de desarrollo

Para el desarrollo del proyecto se ha intentado seguir el modelo SCRUM con prototipos. Este pretende simular la metodología ágil del desarrollo software sobre el hardware. Los prototipos con la piedra angular del proceso y hacia ellos van dirigidos todos los sprint. Así, una vez desarrollados los prototipos básicos, podremos ir reutilizando los y extendiéndoles mejoras sin necesidad de volver a diseñar. Abaratando costes y reduciendo tiempos de producción.

Esta sería la idea fundamental, el modelo SCRUM define nuestra forma de trabajar y organizarnos y el desarrollo de prototipos los incrementos sobre el proyecto. En las etapas iniciales se parecerá mucho al SCRUM, y en las finales al desarrollo de prototipos. Esta idea en software podría decirse que se parece a la programación extrema (desarrollo de prototipos).

5.2. Métodos, herramientas y técnicas

La metodología en las funcionalidades hardware, todo lo referente a dispositivos, se basa mucho en la estrategia de prueba y error. Mientras que en la aplicación y el servidor se buscará siempre programación eficiente y clara, conforme a los documentos de estilo que se elaboren.

Capítulo 6

Planes de soporte del proceso

6.1. Plan de evaluación

Las tareas de evaluación se realizarán gracias a las colaboraciones de amigos y familiares que actuarán como usuarios, asemejando los usos reales de la aplicación. Estas evaluaciones se llevarán a cabo tras la validación de los test asociados al prototipo. Además la evaluación se desarrollará sin la intervención de los participantes del proyecto. Simulando una atmosfera de realidad. Siempre, buscando recabar los datos más fiables posibles sobre el uso de la aplicación.

6.2. Plan de documentación

Durante la elaboración, desarrollo y mejoras de las funcionalidades, se generarán manuales básicos en vista a las evaluaciones. Estos manuales deberán incluir, siempre que se pueda, imágenes o capturas de las acciones a realizar por el usuario. Dirigiéndonos siempre a un usuario medio.

Anexo complementario

Adjuntamos a la documentación el plan de proyecto generado con Microsoft Project