Resumen primera entrega de Sprint 1

Resumen de las User Stories:

Objetivos del Sprint:

Realizar del Product Backlog las siguientes User Stories:

|  |  |  |
| --- | --- | --- |
| **AssetType** | **Name** | ID |
| Story | Investigar, testear características de librerías de Computer Vision | S-01001 |
| Story | Investigar testear sobre librerías y Drivers para MS Kinect. | S-01002 |
| Story | Investigar y desarrollar prototipos sobre librerías de captura y filtrado de imágenes. | S-01003 |
| Story | Desarrollar, documentar e implementar el módulo de captura de imágenes. | S-01004 |

Estimación y prioridades:

|  |  |  |  |
| --- | --- | --- | --- |
| **AssetType** | **Name** | Estimate | Priority |
| Story | Investigar, testear características de librerías de Computer Vision | 24 | High |
| Story | Investigar testear sobre librerías y Drivers para MS Kinect. | 8 | High |
| Story | Investigar y desarrollar prototipos sobre librerías de captura y filtrado de imágenes. | 16 | Medium |
| Story | Desarrollar, documentar e implementar el módulo de captura de imágenes. | 64 | High |

Tiempo Real Invertido y porcentaje realizado:

|  |  |  |  |
| --- | --- | --- | --- |
| **AssetType** | **Name** | Real | %Acomp |
| Story | Investigar, testear características de librerías de Computer Vision. | 22 | 100 |
| Story | Investigar testear sobre librerías y Drivers para MS Kinect. | 9 | 100 |
| Story | Investigar y desarrollar prototipos sobre librerías de captura y filtrado de imágenes. | 28 | 70 |
| Story | Desarrollar, documentar e implementar el módulo de captura de imágenes. | 37 | 50 |

**User Story: Investigar, testear características de librerías de Computer Vision**

**ID:** S-01001

**Nombre:** Investigar, testear características de librerías de Computer Vision.

**Objetivo:** Conocer sobre las librerías de Computer Vision, e identificar cuáles son las más funcionales y seleccionar la que más contemple los requerimientos del proyecto.

**Cumplimiento del objetivo:** el objetivo fue cumplido exitosamente y a término. Como resultado se seleccionó la librería OpenCV ya que es aquella provee la mayor de las funcionalidades requeridas para el proyecto y posee una buena documentación para sustentar gran cantidad de pruebas.

**Desarrollo involucrado:** investigación a través de la web de las librerías de Computer Vision y selección de la librería de OpenCV. El detalle de selección

**Breve descripción de la Librería de visión por computador seleccionada:**

OpenCV es una [librería](http://es.wikipedia.org/wiki/Biblioteca_(programaci%C3%B3n)) libre de visión artificial originalmente desarrollada por [Intel](http://es.wikipedia.org/wiki/Intel_Corporation). Desde que apareció su primera versión alfa en el mes de enero de 1999, se ha utilizado en infinidad de aplicaciones. Desde sistemas de seguridad con detección de movimiento, hasta aplicativos de control de procesos donde se requiere reconocimiento de objetos.

Open CV es multiplataforma, existiendo versiones para [GNU/Linux](http://es.wikipedia.org/wiki/GNU/Linux), [Mac OS X](http://es.wikipedia.org/wiki/Mac_OS_X) y [Windows](http://es.wikipedia.org/wiki/Microsoft_Windows). Contiene más de 500 funciones que abarcan una gran gama de áreas en el proceso de visión, como reconocimiento de objetos, calibración de cámaras, procesamiento de imágenes y representación de imágenes.

Esta desarrollada en código C y C++ optimizados, aprovechando además las capacidades que proveen los procesadores multinúcleo.

**Documentos generados:**

DAR:

**Pruebas realizadas:**

* Instalación y configuración de OpenCV bajo Linux y Windows.
* Configuración de entorno de desarrollo Netbeans y Visual Studio con OpenCV.
* Ejecución de ejemplos de prueba de OpenCV.

**User Story: Investigar testear sobre librerías y Drivers para MS Kinect.**

**ID:** S-01002

**Nombre:** Investigar testear sobre librerías y Drivers para MS Kinect.

**Objetivo:** investigar cuales controladores son los mejores que facilitan un desarrollo ágil y funcional empleando a MS Kinect.

**Cumplimiento del objetivo:** el objetivo fue cumplido exitosamente y con un pequeño desfasaje. Como resultado se seleccionó los controladores Kinect SDK, ya que son aquellos proveen los controladores con gran cantidad funcionalidad para realizar la implementación del proyecto y además posee una gran documentación.

**Desarrollo involucrado:** investigación a través de la web de los controladores de MS Kinect y selección de Kinect SDK.

**Breve descripción de los controladores para MS Kinect seleccionados:**

Kinect SDK  son controladores y [librería](http://es.wikipedia.org/wiki/Biblioteca_(programaci%C3%B3n))s desarrolladas por Microsoft para controlar MS Kinect. La misma puede integrarse con el Framework .NET 4.0 en adelante.

Kinect SDK solo funciona en [Windows](http://es.wikipedia.org/wiki/Microsoft_Windows). Contiene gran cantidad de clases de acceso al firmware interno de la Kinect y una gran cantidad de funciones para desarrollar aplicaciones donde es empleada la Kinect sobre MS Windows.

Está orientada para trabajar con placas de video y procesadores de mediano rendimiento. Las implementaciones de codificación pueden realizarse empleando los lenguajes C#, C++ y Visual Basic.

**Documentos generados:**

DAR:

**Pruebas realizadas:**

* Instalación y configuración de Kinect SDK Windows.
* Configuración de entorno de Visual Studio con Kinect SDK, DirectX 11 y XNA Studio.
* Ejecución de ejemplos de prueba de WPF con Kinect SDK.

**User Story: Investigar y desarrollar prototipos sobre librerías de captura y filtrado de imágenes.**

**ID:** S-01003

**Nombre:** Investigar y desarrollar prototipos sobre librerías de captura y filtrado de imágenes.

**Objetivo:** Conocer sobre las librerías de filtrado de Imagen provistas por OpenCV, para identificar cuáles formatos de captura y filtros de imagen son necesarios para realizar captura y preprosesamiento de imágenes.

**Cumplimiento del objetivo:** el objetivo no fue cumplido exitosamente aun. Se probaron tanto filtros bajos como altos para el mejoramiento de imágenes de la librería OpenCV, filtros para detección de bordes y formatos de representación de captura de imágenes. Se pueden en la actualidad aplicar ciertos filtros de mejoramiento de imágenes realizar obtención parcial de los bordes y realizar representación matricial de imágenes en RGB y escala de grises.

**Desarrollo involucrado:** investigación de filtros y métodos de captura de la documentación de OpenCV, desarrollo de pruebas en C++.

**Breve descripción de lo investigado:**

Los filtros de Blurs son filtros pasa bajo para eliminar el ruido de las imágenes.

Los filtros laplacianos , Canny, Sharr y Sobel son filtros pasa altos son para detectar variaciones de luminosidad.

Realiza captura de imágenes en RGB y escala de Gris de imágenes JPG.

**Documentos generados:**

Pruebas realizas: Pruebas de código en OpenCV.

**Pruebas realizadas:**

* Pruebas de bajo Linux realizadas exitosamente.
* Pruebas realizadas de: Laplaciano, Canny, Sharr, Sobel, threshold, Blurs, Matrices de grises y RGB sobre imágenes en formato JPG.

**User Story: Desarrollar, documentar e implementar el módulo de captura de imágenes.**

**ID:** S-01004

**Nombre:** Desarrollar, documentar e implementar el módulo de captura de imágenes.

**Objetivo:** Desarrollar un módulo de captura que integre todas las funcionalidades investigadas.

**Cumplimiento del objetivo:** el objetivo no fue cumplido exitosamente aun. Se realizo un diagrama de clases de la mayoría del módulo de captura.

**Desarrollo involucrado:** desarrollo de diagrama de clases de Diseño del módulo de captura implementación de codificación en C# y C/C++.

**Documentos generados:**

Pruebas realizas: Diagrama de clases de Diseño del Módulo de Captura .Pruebas de código en OpenCV.

**Pruebas realizadas:**

* Pruebas de bajo Linux y Windows realizadas exitosamente.