
Equipo 2

**Recolector automático de evidencias (AEC)
Documento de Arquitectura**

Version 1.0

Revision History

Date	Version	Description	Author
<29/08/2022>	<1.0>	1ra versión del documento	Randy Reynosa

Table of Contents

Introducción	4
Propósito	4
Alcance	4
Definiciones, Acrónimos y Abreviaturas	4
Referencias	4
Representación Arquitectónica	4
Objetivos Arquitectónicos y Restricciones	4
Vista de Casos de Uso	5
Use-Case Realizations	5
Vista Lógica	6
Vista General	6
Paquetes de Diseño Arquitectónicamente Significativos	7
Vista de Procesos	9
Vista de Deployment	10
Vista de Implementación	10
Visión General	10
Capas	11
Tamaño y Desempeño (Performance)	11
Calidad	11

Documento de Arquitectura del Software

1. Introducción

La introducción de este documento presenta un resumen de todo el documento de Arquitectura de Software. Incluye el propósito, alcance, definiciones, acrónimos, abreviaciones, referencias y el resumen del documento de Arquitectura de Software.

1.1 Propósito

Este documento proporciona una descripción general completa de la arquitectura, usando diferentes puntos de vista arquitectónicos para explicar los aspectos del sistema. Su objetivo es capturar y transmitir las decisiones arquitectónicas significativas que se han hecho en el sistema.

1.2 Alcance

El documento plantea una estructura y la manera en la que esta estructura define las bases y los componentes que se necesitan para realizar el diseño y funcionalidad del Software. Por ello, este documento definirá aquellos elementos importantes que generan la calidad y el valor del producto.

1.3 Definiciones, Acrónimos y Abreviaturas

- AEC : Automatic Evidence Collector
- UABC : Universidad Autónoma de Baja California

1.4 Referencias

- [*CSPS Software Architecture Document 1.0 \(uhcl.edu\)*](#)
- [*CSPS Examples: Development Case \(uhcl.edu\)*](#)
- [*Template: Software Architecture Document \(uhcl.edu\)*](#)

2. Representación Arquitectónica

Las diferentes vistas de esta arquitectura plantean la estructura y la comunicación que sucede entre ellas.

- *La vista de casos de uso contiene diagramas de casos de uso representados por actores que definen las acciones que dichos actores pueden realizar, describiendo los escenarios que se pueden presentar.*
- *La vista lógica contiene diagramas de paquetes y diagramas de clases para representar dichos paquetes, describiendo una estructura más sólida del sistema.*
- *La vista de procesos utiliza un diagrama similar al que contiene la vista lógica para representar el proceso que administra las tareas que realiza el software.*
- *La vista de despliegue presenta un diagrama de componentes, describiendo aquellos componentes físicos necesarios para el software, como las conexiones a los servidores de Google.*
- *La vista de implementación define las diferentes capas que el sistema presenta, dando una pequeña descripción de qué es lo que contiene cada una de ellas y las opciones disponibles.*

3. Objetivos Arquitectónicos y Restricciones

Dentro de la arquitectura tenemos algunos requisitos y restricciones que influyen en el sistema.

- *El software muestra a los alumnos y los atributos de egreso. La interfaz debe de ser capaz de soportar y mostrar a todos los alumnos.*
- *El software eventualmente tendrá que ser desarrollado por otras personas, por lo que esta arquitectura permite un entendimiento del software.*
- *El funcionamiento debe de ser capaz de aguantar a la alta demanda de docentes.*
- *El software debe de poder ser utilizado a través de un navegador web con acceso a internet.*
- *El medio de almacenamiento es la nube por medio de Google Drive, con espacio ilimitado, por lo que no hay restricciones con respecto al tamaño de almacenamiento.*
- *Los datos de los docentes y los alumnos deben de estar protegidos y solo accesibles para personal docente.*
- *Se deben de considerar los tiempos de carga para la arquitectura que se está realizando.*

4. Vista de Casos de Uso

Describe el conjunto de escenarios y/o casos de uso que tienen una arquitectura sustancial o que enfatizan o ilustran un punto específico y delicado de la arquitectura.

Los casos de uso en este sistema se enumeran a continuación. Una descripción de estos casos de uso se puede encontrar más adelante en esta sección.

- Subir Evidencias
- Observar Evidencias
- Aprobar evidencias

4.1 Use-Case Realizations

Los siguientes diagramas describen los casos de uso del sistema.

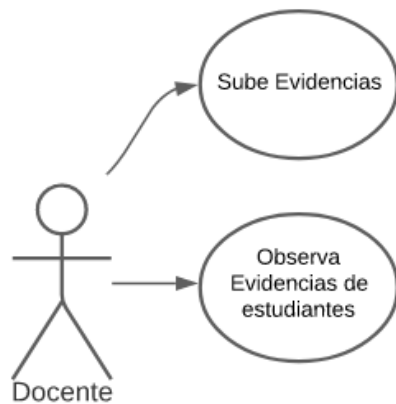


Fig.1 Casos de uso Docente

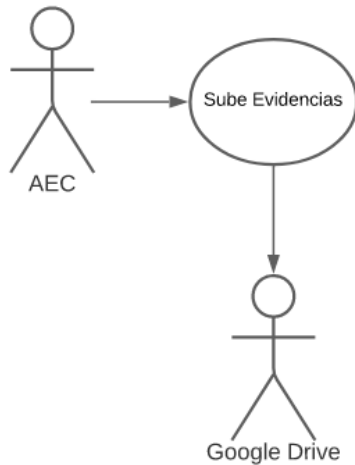


Fig. 2 Casos de uso AEC

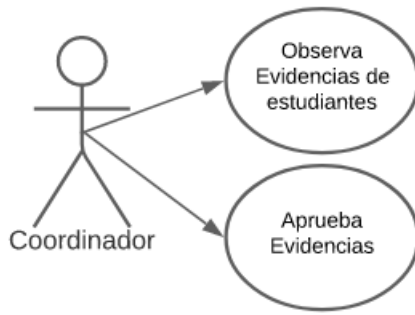


Fig. 3 Casos de uso Coordinador

Descripciones de los casos de usos

1. Sube evidencias

Este caso ocurre cuando un docente o el software sube la evidencia. Dicha evidencia es subida cuando es evaluada por medio de una rúbrica, de igual manera la rúbrica se sube para identificar cómo se evaluó y a qué atributo de egreso corresponde.

2. Observa evidencias de estudiantes

Este caso de uso ocurre cuando el coordinador o docente visualizan la evidencia, dicha evidencia es única para cada alumno.

3. Aprueba evidencias

Este caso de uso toma lugar cuando el coordinador revisa la evidencia y determina si dicha evidencia cumple o no con los lineamientos planteados.

5. Vista Lógica

Describe las clases más importantes, su organización en paquetes de servicios y subsistemas, y la organización de estos subsistemas en capas. Se utilizan diagramas de clase para ilustrar las relaciones entre clases, subsistemas, paquetes y capas arquitectónicamente significativas.

5.1 Vista General

A continuación se muestran los paquetes:

- *Presentación*

Contiene las clases para cada una de las formas que utilizan los actores para comunicarse con el Sistema, como las interfaces.

- *Aplicación*

Contiene las clases que manejan el funcionamiento con el sistema, llevan un control de los procesos y el procesamiento de la información.

- *Dominio*

Contiene la información que necesita el sistema para funcionar como los datos de los alumnos, materias, etc., y presenta clases para el soporte del sistema.

- *Servicios*

Contiene clases para proporcionar mantenimiento al sistema.

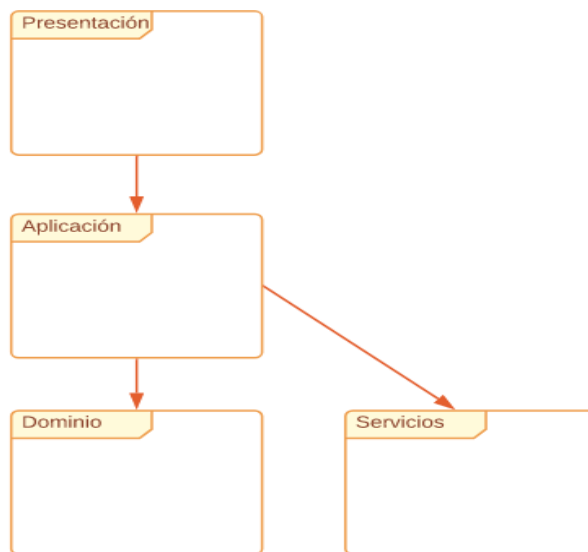


Fig. 4 Vista de paquetes

5.2 Paquetes de Diseño Arquitectónicamente Significativos

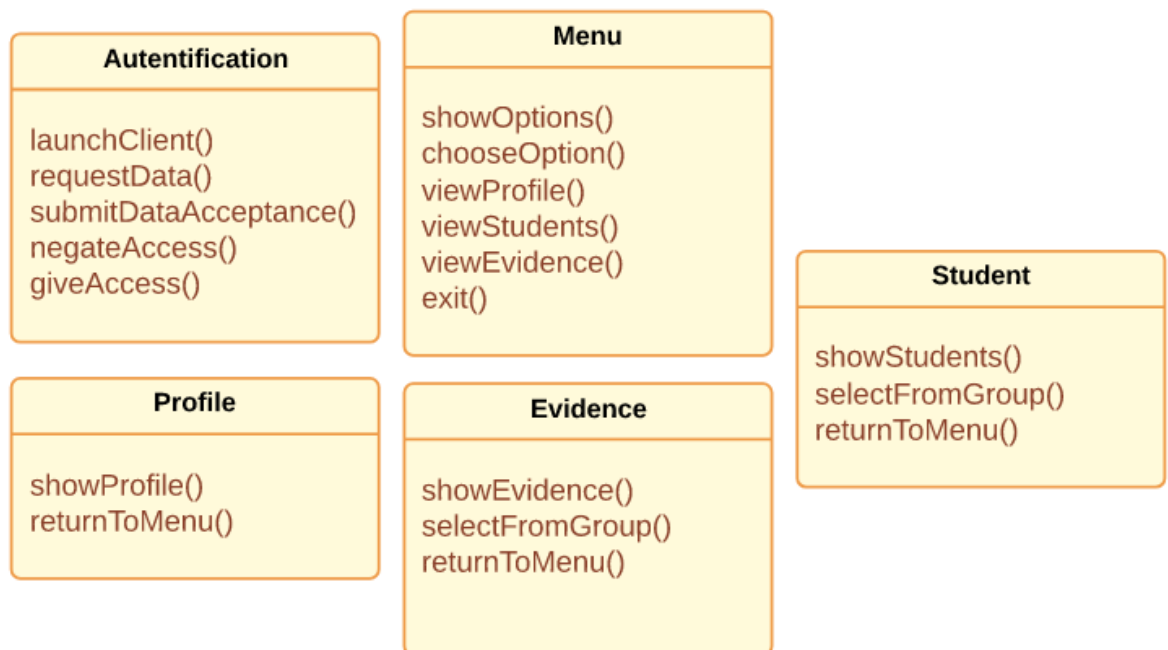


Fig.5 Paquete Presentación

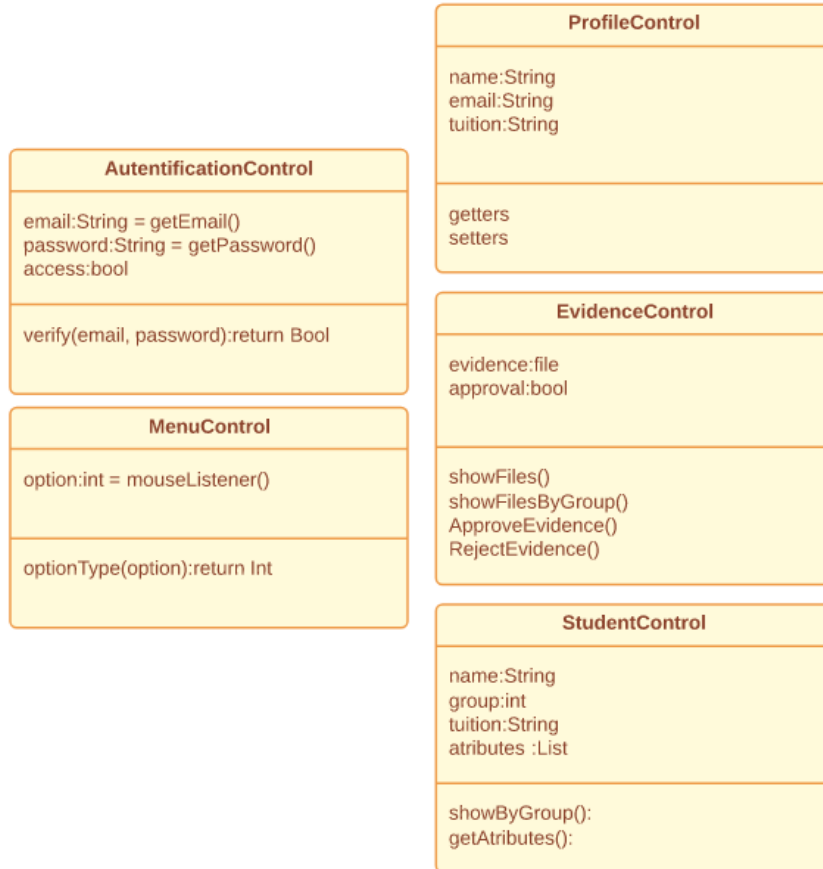


Fig. 6 Paquete Aplicación

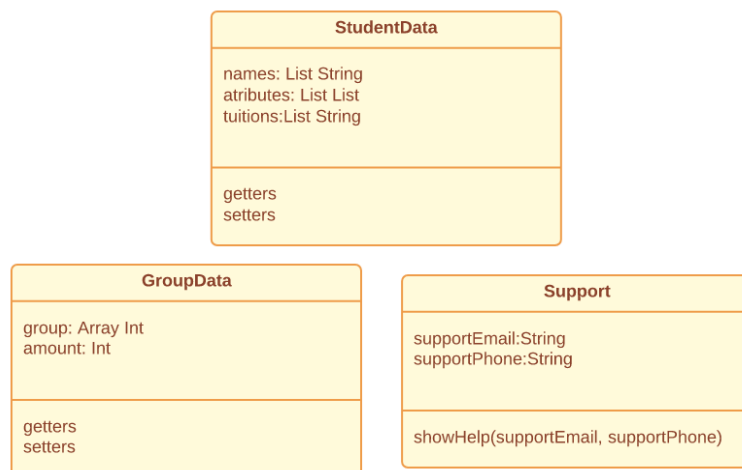


Fig. 7 Paquete Dominio

6. Vista de Procesos

Esta sección describe la descomposición del sistema en procesos ligeros y procesos pesados.

En este punto del diseño, se provee un único proceso para proporcionar funciones a nivel de servidor para el sistema de recolector de evidencias. Los subprocesos para las funciones de la aplicación serán parte de este proceso . El diagrama de proceso del sistema se puede ver de la siguiente manera:

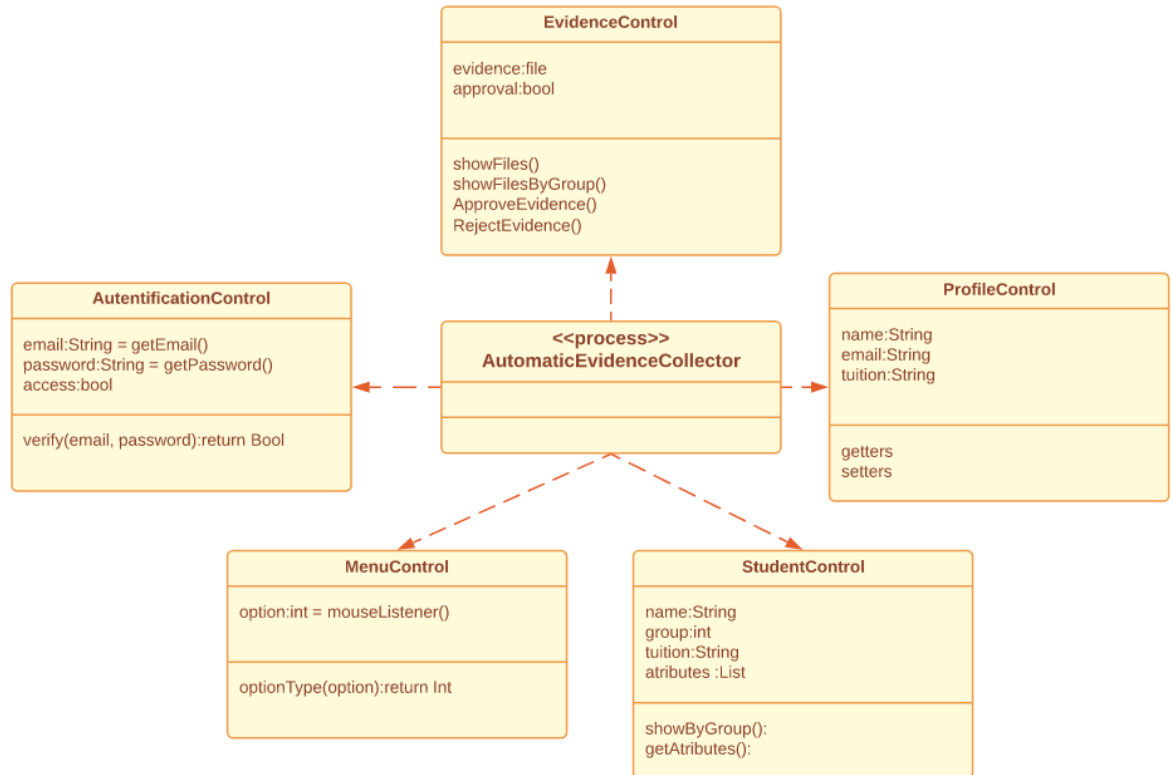


Fig.8 Vista de Procesos

7. Vista de Deployment

Esta sección describe una o más configuraciones de red física (hardware) en las que se implementa y ejecuta el software.

El cliente se conecta por internet a los servidores de Google para acceder a su cuenta, y a su vez, los servidores de Google almacenan la información obtenida y calificada por un API y es almacenada en Google Drive.

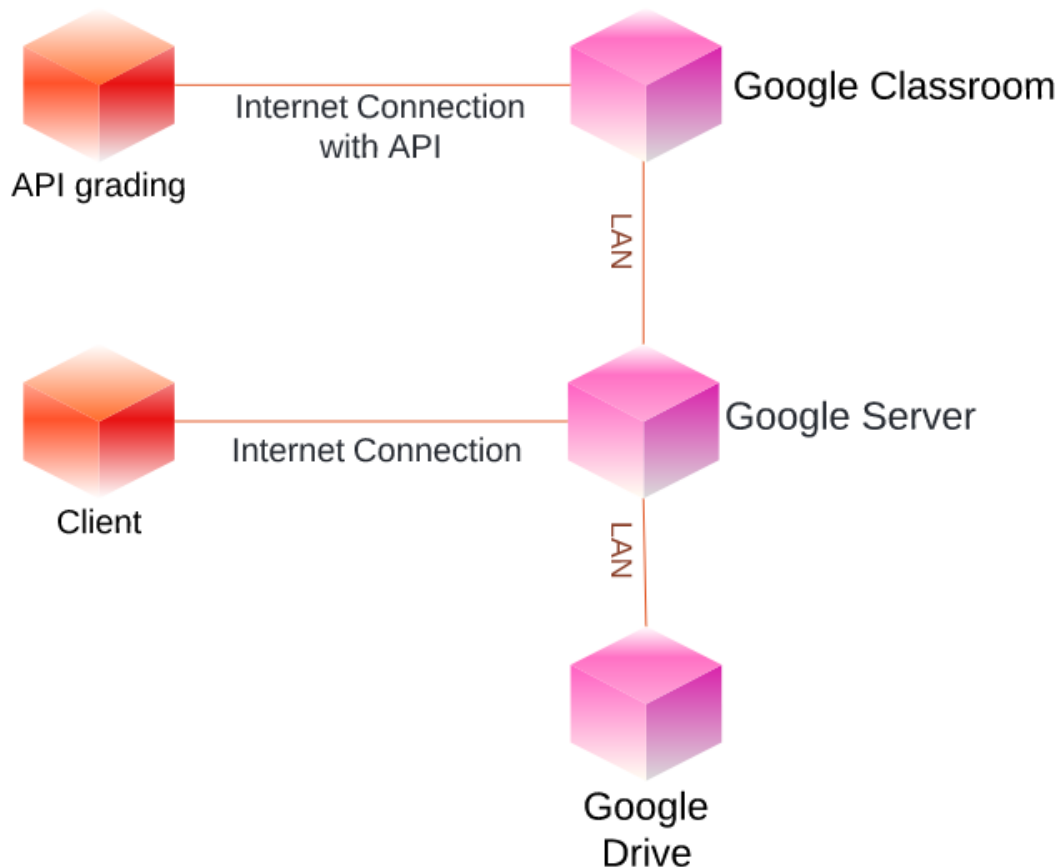


Fig. 9 Vista de Deployment

8. Vista de Implementación

Esta sección describe la estructura general del modelo de implementación, la descomposición del software en capas y subsistemas en el modelo de implementación y cualquier componente arquitectónicamente significativo.

8.1 Visión General

La primera capa presenta una interfaz para un inicio de sesión, una vez verificado y que haya sido autorizado se presenta una capa para mostrar las diferentes opciones que se pueden realizar como:

- Perfil
- Visualizar Evidencia
- Subir Evidencia
- Visualizar estudiantes
- Salir

8.2 Capas

- *Inicio*
Muestra una interfaz que permite ingresar los datos asociados a la cuenta de Google para acceder al software.
- *Menú*
Muestra una interfaz con aquellas opciones que se puedan realizar.
- *Perfil*
Muestra una interfaz con aquellos datos pertenecientes al docente o coordinador de la cuenta.
- *Visualizar Evidencia*
Permite visualizar la evidencia recolectada.
- *Subir Evidencia*
Permite subir evidencia que se considere necesaria realizar de manera manual.
- *Visualizar estudiantes*
Permite dar seguimiento a los estudiantes mostrando los atributos de egreso que estos contienen.

9. Tamaño y Desempeño (Performance)

El software, tal como fue diseñado, soportará a la comunidad docente de manera simultánea. La escala más allá de este nivel se puede lograr proporcionando más servidores que permitan conectar a más docentes.

10. Calidad

El software estará conectado con algunos de los servicios de Google, como Gmail, Google Classroom y Google Drive. Presentará un inicio de sesión a través de Google para identificar al personal docente. El software es compatible con cualquier navegador que permita utilizar los servicios de Google. y provee una interfaz única sencilla para conectar con dichos servicios.