HAC Integration Layer

Vision

# Introduction

# Positioning

## Problem Statement

Atualmente, a plataforma COMPaaS possui a necessidade de criação de uma abstração para cada dispositivo que pretende se comunicar à ela, tornando o processo de adoção do COM- PaaS oneroso neste aspecto. Para eliminar esta necessidade, propõe-se a adaptação do sistema HellfireOS, que já possui a infraestrutura necessária para suportar múltiplos tipos de sensores, para que suporte a comunicação com a plataforma COMPaaS de forma praticamente trans- parente do ponto de vista do suporte de hardware dos sensores ligados ao sistema, permitindo que os produtos que adotem o HellfireOS como seu sistema embarcado estejam aptos a se integrarem à solução IoT de forma mais rápida e simples.

## Product Position Statement

O propósito final do produto, é atender às empresas que pretendem desenvolver sensores e demais hardwares utilizando a inteligência implementada no framework COMPaaS. Como descrito anteriormente, os desenvolvedores que utilizam sistemas de tempo real, atualmente necessitam criar uma camada de abstração compatível com o framework, problema que será contornado com a utilização do HellfireOS em conjunto com o HAC Integration Layer.

# Stakeholder Descriptions

## Stakeholder Summary

| **Name** | **Description** | **Responsibilities** |
| --- | --- | --- |
| [Name the stakeholder type.] | [Briefly describe the stakeholder.] | [Summarize the stakeholder’s key responsibilities with regard to the system being developed; that is, their interest as a stakeholder. For example, this stakeholder:  ensures that the system will be maintainable  ensures that there will be a market demand for the product’s features  monitors the project’s progress  approves funding  and so forth] |

## User Environment

[Detail the working environment of the target user. Here are some suggestions:

Number of people involved in completing the task? Is this changing?

How long is a task cycle? Amount of time spent in each activity? Is this changing?

Any unique environmental constraints: mobile, outdoors, in-flight, and so on?

Which system platforms are in use today? Future platforms?

What other applications are in use? Does your application need to integrate with them?

This is where extracts from the Business Model could be included to outline the task and roles involved, and so on.]

# Product Overview

## Needs and Features

[Avoid design. Keep feature descriptions at a general level. Focus on capabilities needed and why (not how) they should be implemented. Capture the stakeholder priority and planned release for each feature.]

|  |  |  |  |
| --- | --- | --- | --- |
| **Need** | **Priority** | **Features** | **Planned Release** |
|  |  |  |  |

# Other Product Requirements

[At a high level, list applicable standards, hardware, or platform requirements; performance requirements; and environmental requirements.

Define the quality ranges for performance, robustness, fault tolerance, usability, and similar characteristics that are not captured in the Feature Set.

Note any design constraints, external constraints, assumptions or other dependencies that, if changed, will alter the **Vision** document. For example, an assumption may state that a specific operating system will be available for the hardware designated for the software product. If the operating system is not available, the **Vision** document will need to change.

Define any specific documentation requirements, including user manuals, online help, installation, labeling, and packaging requirements.

Define the priority of these other product requirements. Include, if useful, attributes such as stability, benefit, effort, and risk.]

|  |  |  |
| --- | --- | --- |
| **Requirement** | **Priority** | **Planned Release** |
|  |  |  |