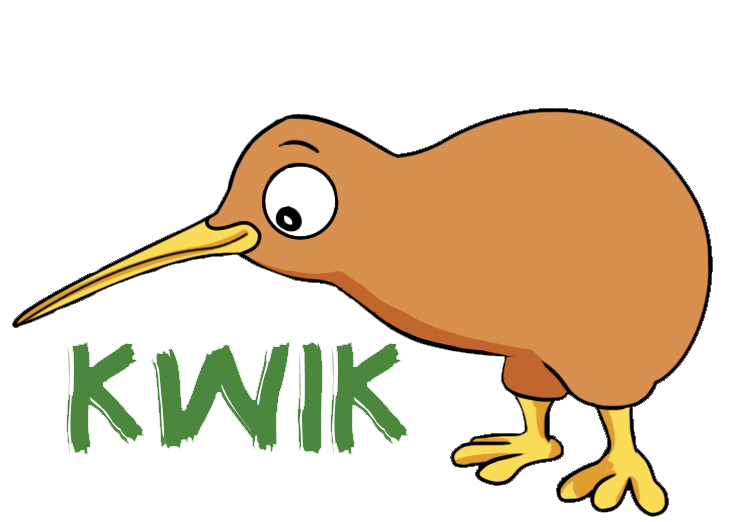
**KWIK**

**Software Architecture Document**

**Version <0.1>**



**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <5/10/2017> | <0.1> | <details> | <name> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Definitions, Acronyms, and Abbreviations 3

1.4 References 3

1.5 Overview 3

2. Architectural Representation 3

3. Architectural Goals and Constraints 3

4. Use-Case View 3

5. Logical View 3

5.1 Overview 3

5.2 Architecturally Significant Design Packages 3

5.3 Use-Case Realizations 3

6. Process View 3

7. Deployment View 3

8. Implementation View 3

8.1 Overview 3

8.2 Layers 3

9. Data View (optional) 3

10. Size and Performance 3

11. Quality 3

**Software Architecture Document**

# **Introduction**

## **Purpose**

This document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

## **Scope**

The Software Architecture Document outlines the direction of development and design specifications which will conform the project.

## **Definitions, Acronyms, and Abbreviations**

## **References**

## **Overview**

# **Architectural Representation**

The project views are composed of the following:

* Home Screen
* Search page
* Cart page
* Purchase page
* User profile

# **Architectural Goals and Constraints**

We require maximum security priority in all areas concerned with monetary transactions including but not limited to credit cards and bank transactions.

# **Use-Case View**

* UC1 – User registration
* UC4 – Article searching
* UC6 – Adding article to cart
* UC9 – Confirmation edition and deletion of articles

# **Logical View**

The project will be developed using MVC development strategy. Within the model there will be three main classes which will be the cart the product and the user. These classes will have other supporting classes such as credit card for the cart or doc type for user which will clarify code and follow OOP conventions.

The system will also have a view which will contain all the visual elements that are displayed in the page.

Finally there is the controller which will handle all the interactions between the view and the model. It will also be in charge of maintaining the database updated when changes are being made to the model

## **Overview**

## **Architecturally Significant Design Packages**

The system will de developed utilizing a MVC oriented development strategy, thus Model View and Controller will be separate from each other within their own packages.

## **Use-Case Realizations**

# **Process View**

The main control flow of the program is derived by Object Messages and exceptions via the Java Virtual Machine. The program will maintain a single-thread architecture.

# **Deployment View**

The software will be able to run through a 32-bit or 64-bit installation of the Java Virtual Machine interconnected to a SQL database through a remote database protocol.

# **Implementation View**

The software will be written in the Java Programming Language, Swing and SQL.

## **Overview**

## **Layers**

# **Size and Performance**

The system's bottleneck will lie between the database and program communication system.

# **Quality**

Reliability will be a primary concern in the development of the KWIK system, since our customers will interact with sensible information such as money and credit cards. A special focus will be made in user centered design in order to increase usability and present a product oriented to satisfy the users demands.