Ikindo WebEdit
Software Requirements Specification

Version 1.0

Revision History

Date	Version	Description	Author
18.10.2020	1.0	Set up template and filled in project name	David
20.10.2020	2.0	Filled out all required sections	David, Jonas, Moritz

Table of Contents

Software Requirements Specification

1. Introduction

This document contains an overall description and specifies all software requirements for the Ikindo WebEdit project. All entries are subject to change during development. This document will be versioned using git and can be found on https://github.com/IkindoWebEdit/ikindo-docs.

1.1 Purpose

The purpose of this SRS is to act as a developer/designer guideline in later stages of development, and to create boundaries between necessary and optional features. This SRS will describe functional and non-functional requirements as well as critically important design decisions.

1.2 Scope

This Project will be realized as a website. The website we want to offer our clients should be an editable layout the client can interact with, through an administrator account. In the end the client should be just able to edit his website without any knowledge about programming and website hosting. The content, shown to normal visitors of the website, can be easily set by the owner in a separate editor site.

1.3 Definitions, Acronyms, and Abbreviations

SRS	Software Requirement Specification
HTML	Hypertext Markup Language
Admin	Site Administrator

1.4 References

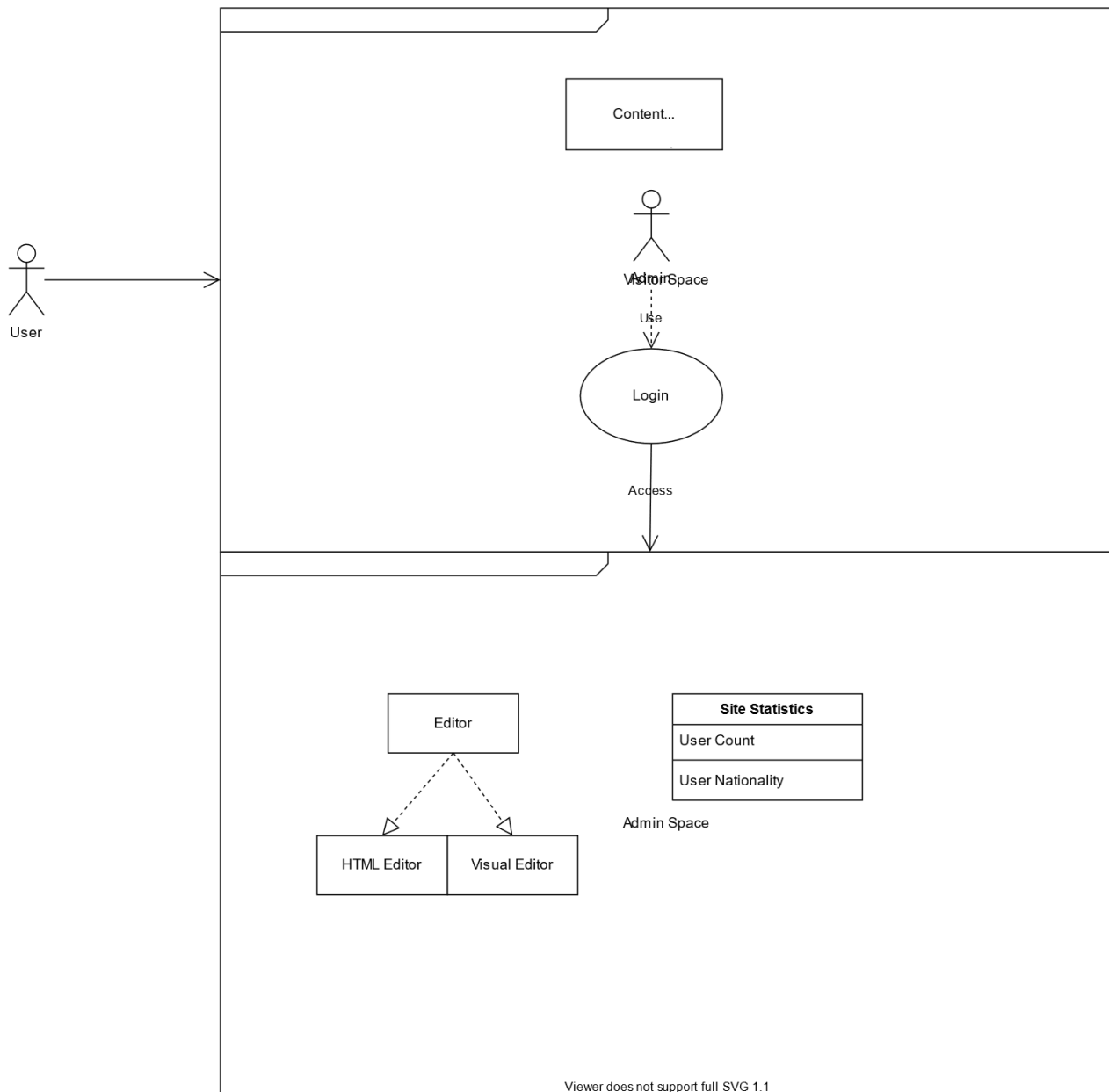| Website	https://ikindowebedit.wordpress.com/
GitHub	https://github.com/IkindoWebEdit

1.5 Overview

In the following paragraphs, we will describe in detail what the exact use cases of our project are, what requirements need to be fulfilled to handle these cases and what design choices were made for the necessary components.

2. Overall Description

The main focus of the project design will be ease of use. Since most users of our web editor will not be very adept at creating websites, the editor should be able to do most of the work for them. Ideally, users will be able to create a website of their liking without ever having to touch the actual HTML source. For more experienced users, the features might feel more limiting than anything else, therefore the admin should also be able to view and edit the source directly from the website.

2.1 Use-Case Diagram

Ikindo WebEdit Use Case Diagram

Content...

User

Visitor Space

Admin Space

Use

Login

Access

Editor

Site Statistics
User Count
User Nationality

HTML Editor | Visual Editor

Admin Space

Viewer does not support full SVG 1.1

2.2 Assumptions and Dependencies

This SRS generally assumes that all ideas and requirements are actually feasible with the tools that we want to use, which might not hold true.

3. Specific Requirements

3.1 Use-Case Reports

1) User accesses the website:

The user will be greeted with the front page of the website.

2) **The user is an admin and actuates the login button:**
   The user will be prompted for a username and a password.
3) **The admin successfully logged in:**
   The admin is presented with a choice to view the website either as-is or as a HTML source document. He can also access the website statistics overview.
4) **The admin chose to view the website as-is:**
   The as-is view represents the graphical editor. The admin can now modify the design of the website and add/remove content using a graphical interface.
5) **The admin chose to view the website HTML source:**
   The source can be edited, and the modified page can be uploaded to the server to replace the current website.
6) **The admin chose to view the site statistics overview:**
   The admin gets presented with useful data.
7) **The admin logs out:**
   The admin loses his admin privileges and can view his changes out of the eyes of a regular user.

3.2 Supplementary Requirements

- To improve ease of use, the graphical editor should clearly show in one overview all tools that the user has at their disposition.
- The website should have a front page which is limited in its editing capabilities, so that the user always has at least one working page to start from.
- The front page always includes an agenda that indexes all subsites of the website. This makes it possible to navigate through all existing pages.

4. Supporting Information