

Assignment 9

Name: Yutong Zhu

Individual Requirements Analysis

Introduction

Project Name

- CHAOSS

Purpose

- The purpose of this document is to specify the requirements and details of the software architecture for OpenSourceHealth, which is CHAOSS, a Linux Foundation project focused on creating analytics and metrics to help define community health. The project aims to:

1. Establish standard implementation-agnostic metrics for measuring community activity, contributions, and health
2. Produce integrated open source software for analyzing software community development
3. Build reproducible project health reports/containers

Scope

- This tool is built for technology companies and IT organizations and open software contributors. Basically, this software is for everyone who wants to assess the health and sustainability of the open source software.

Assumption

- It is assumed that the companies/communities/organizations/developers are willing to provide data to the software.

Software Product Overview

System Scope

- OpenSourceHealth will use an automated algorithm to make sustainability predictions for projects. It will be available for both web and mobile applications. The system will let users be able to search the open source project they are interested in and see how healthy and sustainable they are.

System Architecture

- External view of software product
- Internal view of software product

Feature Overview

- Input and manage customer information
- Collect and Analysis data
- Generate reports

System Use

Actor Survey

- Owner/Proprietor

Owner and proprietor are responsible for identifying new clientele, selling the service, and improve the system. This actor will interact with the system to identify the health and sustainability of open source software.

System Feature: View reports

- Customers

Customers are main users of the system. They will be able to search the open source software they are interested in and get report from the system to know about the open source software's health and sustainability. They will make decision based on the reports provided by the system.

System Feature: View reports

- Manager/Employee

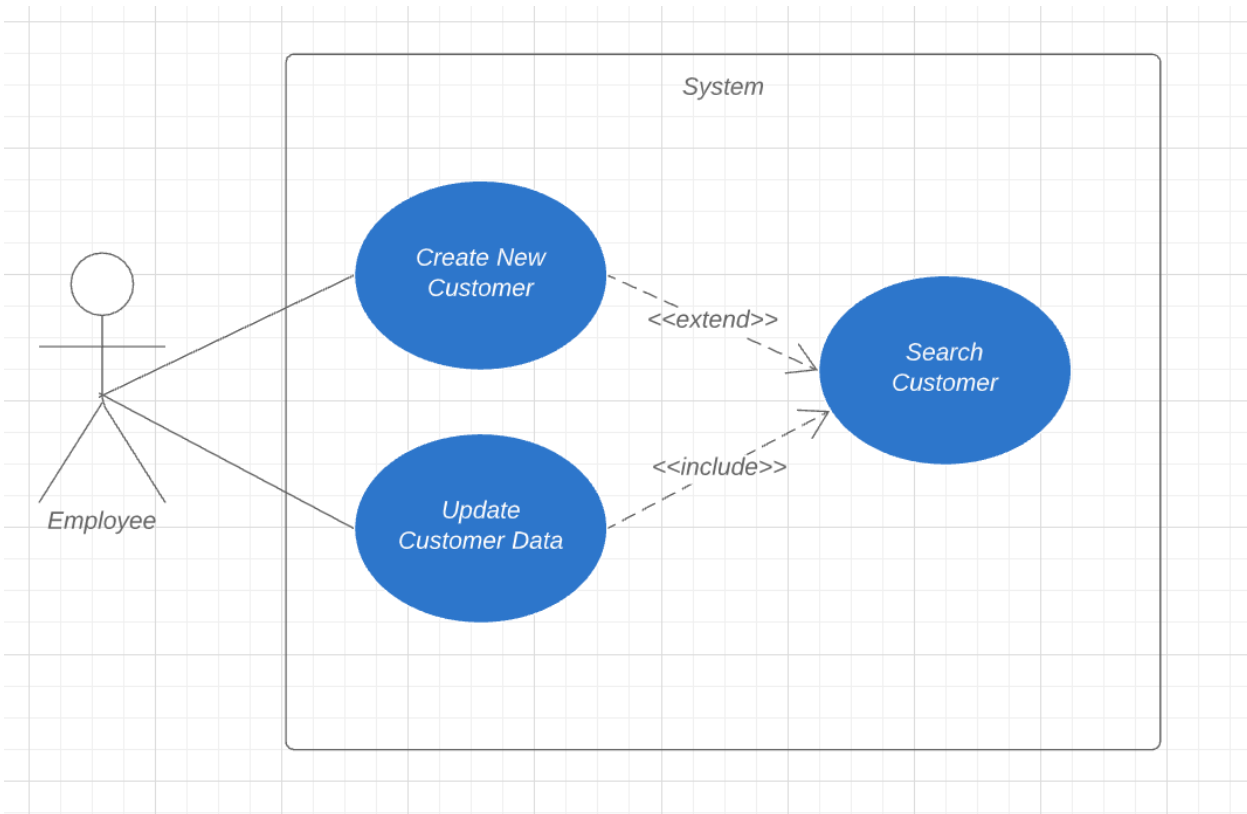
Manager and Employee are the administer of the system. They have the access to the core algorithm and data used in the system. They could enter customers information and improve the algorithm of the system to achieve a better, precise report.

System Feature: Collect and Analysis data; Input and manage customer information

System Requirements

System Use-Cases

Use Case 1: Input and Manage Customer Information



Actors: Employee

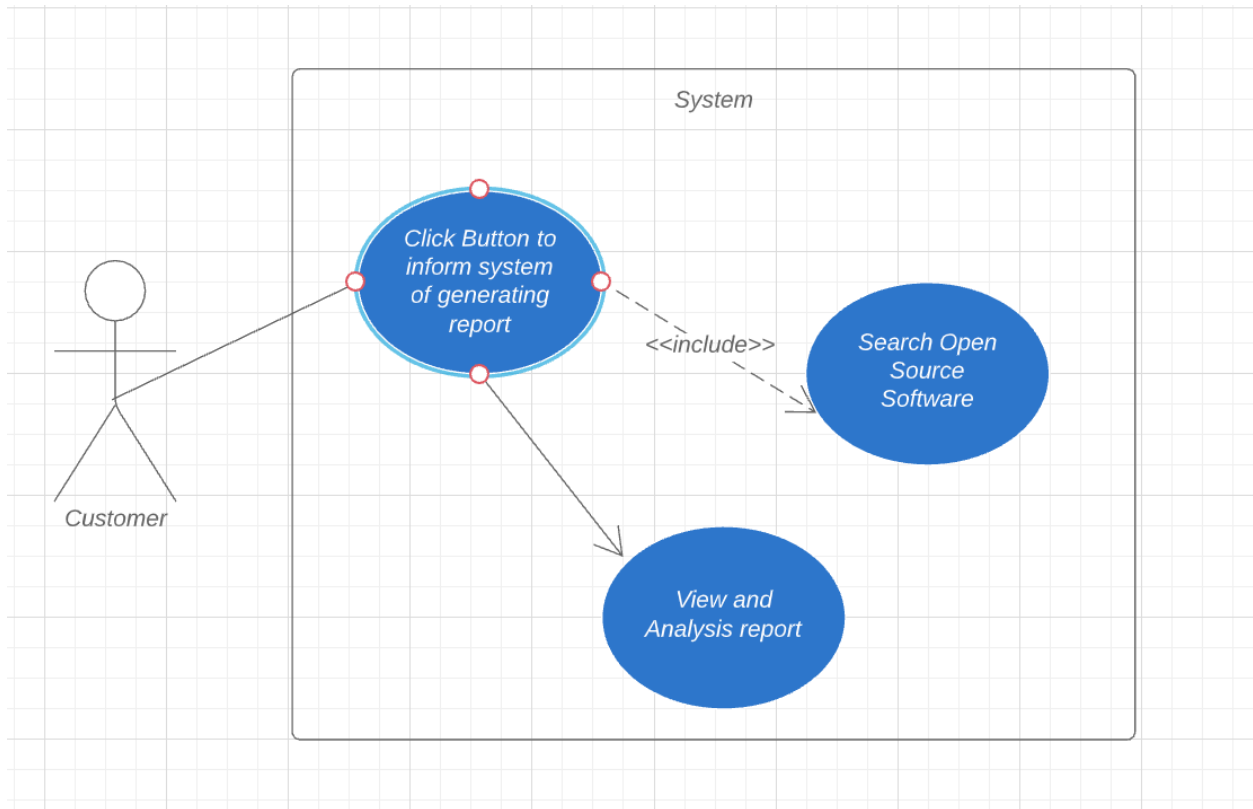
Brief Description: This use case explains how an employee enters and stores all customer information such as name, address, telephone, email and customer's landscaping needs.

Basic flow of events: 1.System displays customer entry; 2.Employee enter customers' info; 3.Save and update system

pre-conditions: The employees must be authenticated to do so

Extension Point: Search customer

- User Case 2: Create Open Source Software Health and Sustainability Report



Actors: Customer

Brief Description: This use case explains how a customer let system generate a report for assessing the health and sustainability of a specific open source software after logging in to the system and searching for a specific open source software

Basic flow of events: 1.Customer log in to the system; 2.Customer search for a specific open source software 3.System generate report after customer click "generate report" button 4.Customer view the report regarding health and sustainability of the open source software

pre-conditions: Customers have already signed up

System Functional Specification

- The system algorithm of assessing health and sustainability of open source software should be good enough to provide precise prediction
- The system data should be updated in time
- Customers should login first before require reports

Non-functional Requirements

- System should be stable enough to support multiple users requiring reports at the same time
- System should be able to handle big data
- System should have good hardware which support it to assess data in real time
- All system responses shall occur within 30 seconds. interactions.
- User interface is user-friendly

Design Constraints

- The system will run as a web application with access via an encrypted SSL login.
- The application will run on all major web browsers - Internet Explorer, FireFox, Chrome, Safari and their mobile counterparts.
- We will customize key use cases to be available as mobile specific content.
- Each web browser adheres to web standards differently and Prunelt must overcome these differences in order to run seamlessly on any platform.
- The system should use such technologies as Java, Spring, MySQL, Apache and Windows.

Purchased Components

- Software used to develop this application should be open source
- Data used by this application: \$10000 / month
- Web Server/ Database Server: \$5000
- Computers: \$100000

Interfaces

- User interface:
 1. Home page: login / Sign up
 2. Search Page: Search Open Source Software by name
 3. Search Result Page: Show Software basic info and a button for requesting report
 4. Report Page: Show data related to the open source software and assess of the data regarding health and sustainability