Orienteering Event Management System

Overview

The goal of this project is to develop an end-to-end orienteering event management system for orienteering event organisors. This is composed of the following:

- Competitor registration
- Course registration
- Event data manager
- Results manager

What Is Orienteering?

Orienteering is a sport involving a mix of physical and technical skills. Competitors must visit various points in a specified order across a variety of terrain such as forest, mountains and sand dunes, aiming to be the fastest person to complete the course. Each point ("control") has an SI Box - a device with a hole in it for the timing chip. Every competitor has an wearable electronic timing chip, called an SI Card, that records a visit to the control. The fastest competitor wins.

Competitor Registration

Competitors in an orienteering event generally use the SI Card, a type of timing chip used by participants to prove that they visited a specified point. There is an public online database that maps a competitor's own SI Card to their name.

The registration system would record which course the competitor would like to visit, their SI Card number, and their name.

Course Registration

At a logical level, each course is an ordered sequence of controls. We would like to build an interface for event managers to define their courses in. This information would be stored in a database so it can be processed elsewhere in the application.

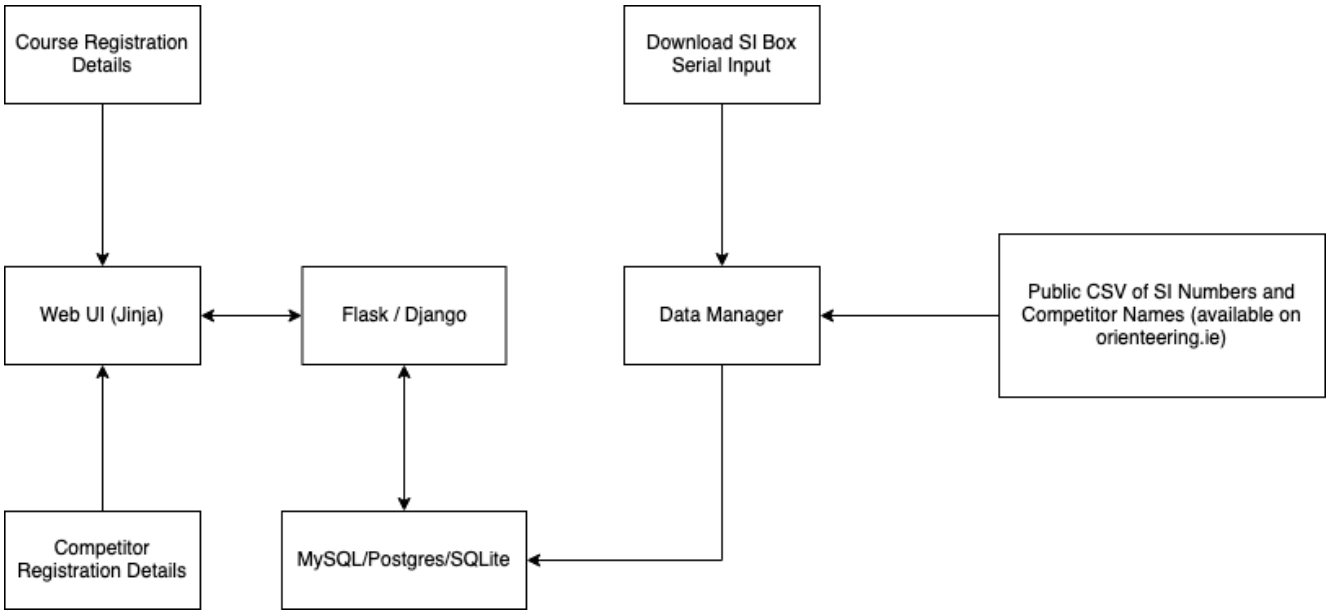
Event Data Manager

When a competitor completes a course, they must punch the "Download" control that is connected by USB to the event manager's laptop. This uploads the data stored on the SI Card to the event manager's computer over serial.

The data is a sequence of bytes containing information such as the competitor SI number, the control number, and the timestamp.

Results Manager

The bulk of this project involves the processing and management of the data that we receive over serial from the device. This is a nontrivial task. Once the data is processed, it will be stored in a database so the web application can interact with it. Below is a diagram that depicts the various relationships between each part of the software.



- Course registration and competitor registration is provided via the Web UI, which may be written using Jinja / React. This would be built on Flask or Django. This information can be uploaded to the SQL database using Python.
- The Flask / Django backend communicates with the database to read and write information as needed.
- The serial input (in bytes) is received by the data manager, which performs operations on this data to clean it up and insert it in the database once it is satisfactorily processed.
- SI numbers will be mapped to the SI numbers provided by the serial input to map names to SI numbers.
- The final, processed data is fetched by Flask to be displayed as needed for the event organisor's convenience.

- The organiser hosts this application locally at the event, and the Download SI Box is connected via USB and the data is communicated via serial this way.