Cooperative Education Site Visit Scheduler

Supplementary Specification (Inception Draft)

Introduction

This document is the repository for all Site Visit Scheduler requirements not captured in the use cases.

Functionality

Log all transactions to persistent storage for review by the System Administrator and possible recovery from prematurely terminated activities.

Usability

The system must be usable by Students and Faculty Members with varying levels of computer knowledge and skills. It should be easily navigable and provide clear instruction for relatively naïve users. To the extent possible, it should provide fast methods of data entry for more sophisticated users.

Reliability

Recoverability: The system should log all transactions, and permit the user to recover information later in the event of system
failure. In case of failure of the primary persistent store, the system should place any complete records in a queue for entry into primary persistent storage when the system recovers.

**Performance**

The system should respond to user requests within 1 second 90% of the time.

The system should be available to users at least 95% of the time.

**Supportability**

The system should be maintainable and extensible to allow for changing types of faculty constraints on the site visit itinerary.

**Implementation Constraints**

The primary developer insists on using Java, predicting that this will improve long-term portability and supportability. In addition, various freely available Java development environments will make development easier. (Also, he wants to learn more about Java.)

**Purchased Components**

We do not anticipate the need to purchase any of the components for this system.

**Free Open Source Components**

The project should make maximum use of free open-source components.
Although it is too early to definitively design or select components, the following are worth consideration:

- JLog logging framework
- MySQL database (assuming we decide to use a database for persistent storage)

**Interfaces**

Users should be able to interact with the system via a Java-enabled World Wide Web browser.

**Application-Specific Domain Rules**

<table>
<thead>
<tr>
<th>ID</th>
<th>Rule</th>
<th>Changeability</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>RULE1</td>
<td><strong>Travel time:</strong> Site visit itinerary must allow adequate time between visits for travel from first site to source (MSU), travel from source to second site, extra time specified by faculty member, time for lunch (if applicable) as faculty member specifies</td>
<td>High</td>
<td>Faculty advisor interview</td>
</tr>
<tr>
<td>ID</td>
<td>Rule</td>
<td>Changeability</td>
<td>Source</td>
</tr>
<tr>
<td>------</td>
<td>-------------------------------------------</td>
<td>---------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>RULE 2</td>
<td><em>Travel zones:</em> The faculty member should be able to assign a named zone to each worksite (e.g. “NJ” or “NYC”) as a basis for constraints to ease travel and reduce travel times.</td>
<td>High</td>
<td>Different faculty will specify different zones and use them as a basis for different types of restrictions.</td>
</tr>
</tbody>
</table>
| RULE 3 | *Zone constraints:* Faculty members should be able to specify the following types of constraints:  
- Restrict permissible work sites for a particular day or time to given zone(s)  
- Allow visits to only one (unspecified) zone on any (unspecified) day. | High          | The goal is to allow the faculty member to group sites together for convenience and reduced travel time. There may be a better way to achieve this goal. | Faculty member interview |