**Vision Document for “MIUScheduler”**

**Team members:**

Oussama Jablaoui 610712

Adham Mohamed 610768

Mark Ghatas 610584

Abanoub Kaiser 610738

**1. Introduction**

Several years ago in the Computer Professional MS in CS program, there were three

entries per year and student entry numbers were 20-40 per entry. Often there was just one

elective class being offered per block and all students in an entry took the same classes in

the same sequence. Scheduling of classes and faculty was done with a relatively simple

Excel spreadsheet, and students were assigned to classes via a manual process.

As Compro has grown, we now offer 4 entries per year and there are often 100 – 130

students per entry. In some blocks, we may offer 8 or 9 elective classes, plus there are

often 3 FPP classes and 5 MPP classes offered per entry. There are several areas of

specialization for classes such as:

- Web Applications

- Data Science

- SW Design

- Networking

- Operating Systems

- Compilers

- Parallel Programming, etc.

Most faculties have one or two areas of specialization and a set of classes that they

would like to teach. In addition, they have preferences for what blocks they can teach.

Faculty needs to be able to enter their profile and be able to view their scheduled classes.

Compro students should be able to view the schedule and register for classes.

A few 500 level courses have 400 level **course prerequisites**, so the 400 level courses

should be offered for each entry in their first blocks on campus.

The 500 level classes should be provided for their later blocks on campus.

Most students take 4 elective blocks on campus.

Some U.S. resident students take 9 elective blocks on campus.

Some OPT students take 5 courses on campus.

MIUScheduler is a new software tool that will build a Compro schedule of classes with

faculty assigned to each class and will also offer a simple tool for students to register for

those classes.\*

(\*Note – the student registration part will be kept simple for our project. It is added for

the purposes of having a separate student register subsystem – to be explained in later.)

**2. Positioning**

**2.1 Problem Statement**

|  |  |
| --- | --- |
| The problem of | *managing the Compro schedule and allowing students to*  *register for classes* |
| Affects | *administrators, faculty, and students* |
| the impact of which is | *scheduling is complex, must be manually maintained, and changed frequently* |
| a successful solution would be | *one tool which builds a Compro schedule that integrates the business rules for faculty availability and courses needed by students per entry. This tool will provide a Database and a user interface that is easy to use for faculty, staff, and students.* |

**2.2 Product Position Statement**

|  |  |
| --- | --- |
| For | *administrators, faculty, and students* |
| Who | *Either register for classes or schedule their teaching courses* |
| The MIUScheduler | *Is a tool* |
| That | *Builds the compro schedule while integrating the*  *business rules for faculty availability and courses needed by students per entry* |
| Unlike | *Manually maintaining the schedule* |
| Our product | *Automates the process and provides a user-friendly interface that is easy to use for faculty, staff and students.* |

**3. Stakeholder Descriptions**

**3.1 Stakeholder Summary**

****

**3.2 User Environment**

4 people are involved in completing the feature set, this is fixed and should not change isA, task cycles should be around 3 days long with a range of 10-12 hours spent in total 2-3 hours on each individual assignment. No environmental constraints available in this project, the system platforms to be used will be as follows:

Currently Web based, in the future an android mobile application may be developed.

**4. Product Overview**

**4.1 Product Perspective**

*this product is independent and totally self-contained.*

**4.2 Assumptions and Dependencies**

* *If a new track was added other than FPP or MPP this will affect the scheduling system*
* *If implementation of the prioritization of the students changed this will affect how students get assigned to courses.*

**4.3 Needs and Features**























**4.4 Alternatives and Competition**

*Alternatives might be a legacy Scheduler system or even manual maintenance of course availability and faculty members in a system database. No insights on the legacy scheduler system thus will not be able to list the strengths and weaknesses, however here are a few for the manual maintenance alternative solution:*

*Strengths:*

* *Easy to understand*
* *Lower cost.*

*Weaknesses:*

* *Outdated*
* *Loss of data*
* *Difficult to Protect*
* *Lack of Space and Efficiency*
* *Redundancy*

**5. Other Product Requirements**

*Hardware/platform: any 64 bit OS with JRE 8 and above, x64 based processor*

*Preferably 16GB RAM or above, performance requirements are directly proprotional with the university capacity / number of students. A user manual should be provided with instructions on how to use the web, there should be an online chat to help the students with their inquiries regarding using the tool.*