**Vision Document for “MIUAirBnb”**

**Team members:**

1. **Purushottam Prajapati - 614191**
2. **Sourav Babu Shrestha - 614690**
3. **Saroj K Dev - 614748**

**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.

MUMSched 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**

*[Provide a statement summarizing the problem being solved by this project. The following format may be*

*used:]*

|  |  |
| --- | --- |
| 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**

*[Provide an overall statement summarizing, at the highest level, the unique position the product intends to*

*fill in the marketplace. The following format may be used:]*

|  |  |
| --- | --- |
| For | *[target customer]* |
| Who | *[statement of the need or opportunity]* |
| The (product name) | *is a [product category]* |
| That | *[statement of key benefit; that is, the compelling reason to buy]* |
| Unlike | *[primary competitive alternative]* |
| Our product | *[statement of primary differentiation]* |

*[A product position statement communicates the intent of the application and the importance of the project*

*to all concerned personnel.]*

**3. Stakeholder Descriptions**

**3.1 Stakeholder Summary**

****

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Responsibilities** |
| **Admin** |  |  |
| **Tenant** |  |  |
| **Landlord** |  |  |

**3.2 User Environment**

*[Detail the working environment of the target user. Here are some suggestions:*

*Number of people involved in completing the task? Is this changing?*

*How long is a task cycle? Amount of time spent in each activity? Is this changing?*

*Any unique environmental constraints: mobile, outdoors, in-flight, and so on?*

*Which system platforms are in use today? Future platforms?*

*What other applications are in use? Does your application need to integrate with them?*

*This is where extracts from the Business Model could be included to outline the task and roles involved,*

*and so on.]*

**4. Product Overview**

**4.1 Product Perspective**

*[This subsection of the* ***Vision*** *document puts the product in perspective to other related products and the*

*user’s environment. If the product is independent and totally self-contained, state it here. If the product is a*

*component of a larger system, then this subsection needs to relate how these systems interact and needs to*

*identify the relevant interfaces between the systems. One easy way to display the major components of the*

*larger system, interconnections, and external interfaces is with a block diagram.]*

**4.2 Assumptions and Dependencies**

*[List each factor that affects the features stated in the* ***Vision*** *document. List assumptions that, if changed,*

*will alter the* ***Vision*** *document. For example, an assumption may state that a specific operating system will*

*be available for the hardware designated for the software product. If the operating system is not available,*

*the* ***Vision*** *document will need to change.]*

**4.3 Needs and Features**

*[Avoid design. Keep feature descriptions at a general level. Focus on capabilities needed and why (not*

*how) they should be implemented.]*























**4.4 Alternatives and Competition**

*[Identify alternatives the stakeholder perceives as available. These can include buying a competitor’s*

*product, building a homegrown solution, or simply maintaining the status quo. List any known competitive*

*choices that exist or may become available. Include the major strengths and weaknesses of each competitor*

*as perceived by the stakeholder or end user.]*

**5. Other Product Requirements**

*[At a high level, list applicable standards, hardware, or platform requirements; performance requirements;*

*and environmental requirements.*

*Define the quality ranges for performance, robustness, fault tolerance, usability, and similar*

*characteristics that are not captured in the Feature Set.*

*Note any design constraints, external constraints, or other dependencies.*

*Define any specific documentation requirements, including user manuals, online help, installation,*

*labeling, and packaging requirements.*

*Define the priority of these other product requirements. Include, if useful, attributes such as stability,*

*benefit, effort, and risk.]*