CMPT 2276 - Introduction to Software Engineering

**Requirements Gathering Report**

Jimmy Lau, Sandy Gao, & Quang Nam Ta

horizontal line

# **Interview Background Information**

Raymond Ho is a Hong Kong Secondary School Teacher that has been teaching for 23 years. He uses the faculty scheduling system called *“E-Class”* developed independently by the school.

Questions to interview:

* What faculty management system are you currently using? How was your overall experience with it?
* What functions do you think an ideal faculty management system should have?
* What difficulties or challenges have you encountered when using the existing system?
* What are your expectations or suggestions for the new system?

### **The needs and challenges of secondary school teachers on faculty management system:**

**Interviewer:** Hello, and thanks again for your time to do this interview. We are developing a faculty scheduling system and hope to understand the requirements and the challenges you encountered when using the existing system.

**Raymond:** No problem, I am happy to help.

### **The use of existing system:**

**Interviewer:** What faculty management system are you currently using? How was your overall experience with it?

**Raymond:** Our school is currently using E-Class. It is mainly used for course scheduling, attendance records and grade management. Most functions work pretty well, but some parts are not convenient to use, especially the scheduling part.

### **System functional requirements:**

**Interviewer:** What functions do you think an ideal faculty management system should have?

**Raymond:** The system should be able to generate a course schedule automatically and allow us to adjust it manually. Like quickly finding the appropriate time and classroom that is available if I need to adjust my class due to special circumstances.

I want the system to be recording student’s attendance automatically also so I don’t have to do it manually.

The system should also support different grading methods, like the weight between homework and examination.

A notification system would be also, like sending notifications about course changes and meeting reminders.

Able to check all this information on my phone would definitely be a cherry on top.

### **Difficulties in using existing systems:**

**Interviewer**: Are there any difficulties or challenges you encountered when using the existing system?

**Raymond:** It often conflicts when 2 classes are scheduled at the same time or the room is being booked twice. This ends up needing us to make adjustments manually and takes a lot of time.

There are also too many buttons, so I sometimes get lost in them.

If I want a class to be fixed at a specific time, the system can’t do it.

### **Expectation for new system:**

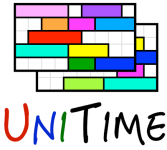
**Interviewer:** What are your expectations or suggestions for the new system?

**Raymond:** Having a simple and clear interface is definitely going to help. Data safety is also important. Having full mobile support is really convenient. And it will be really good if the system can be adjusted to individual needs.

### 

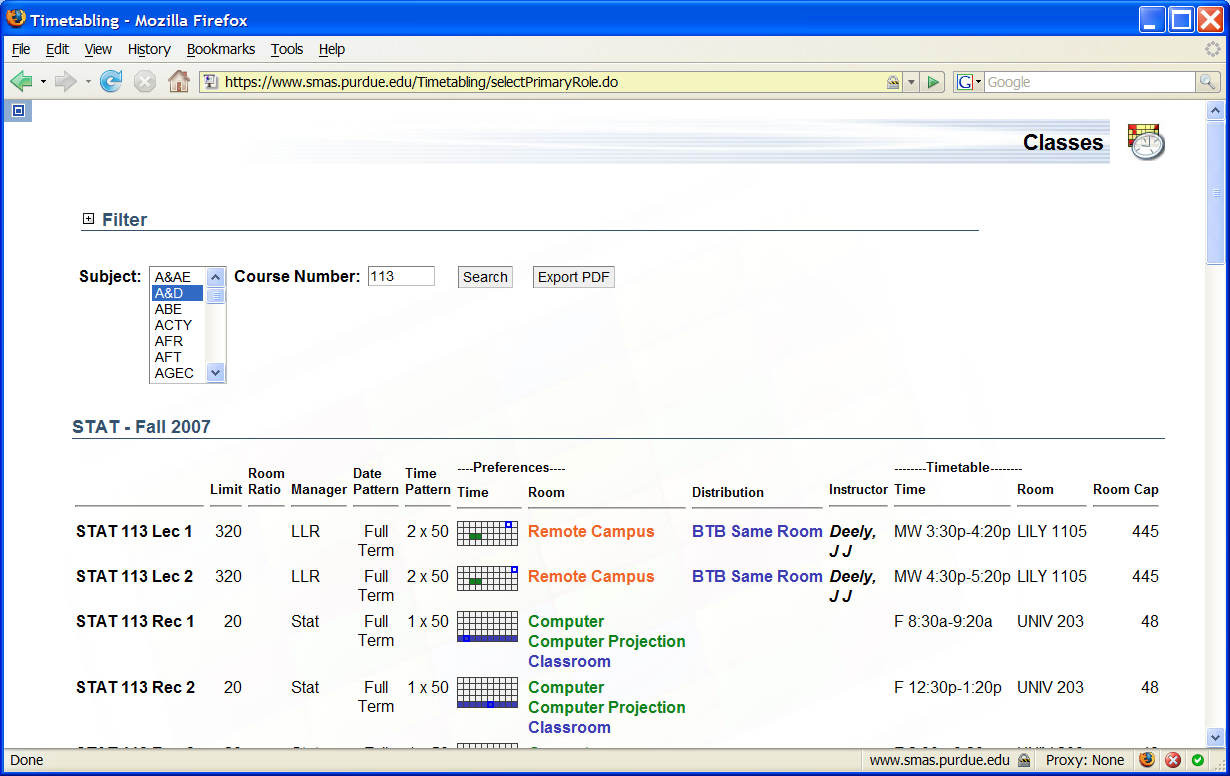
# **Documents Analysis**

### **Background Information**

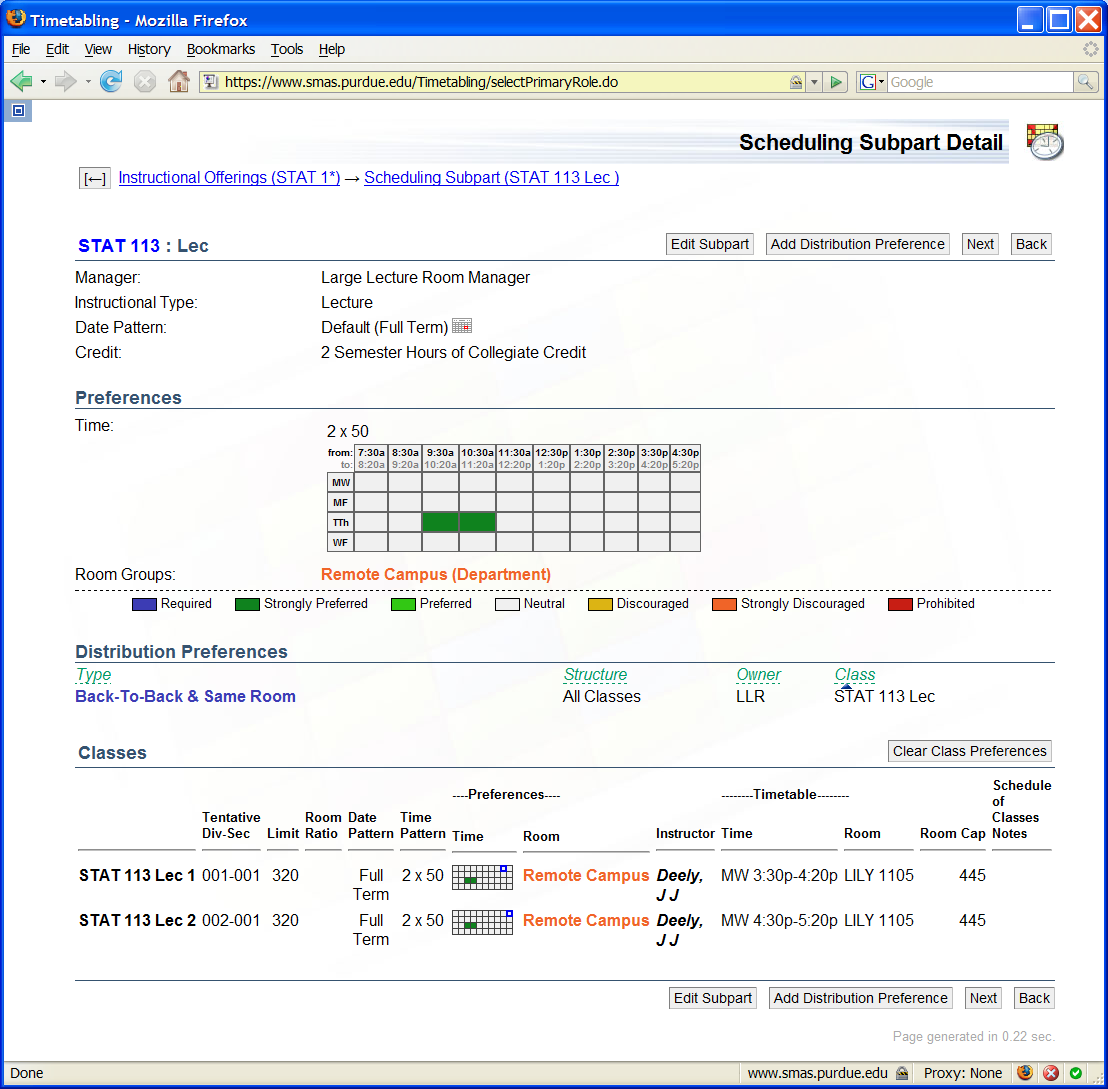


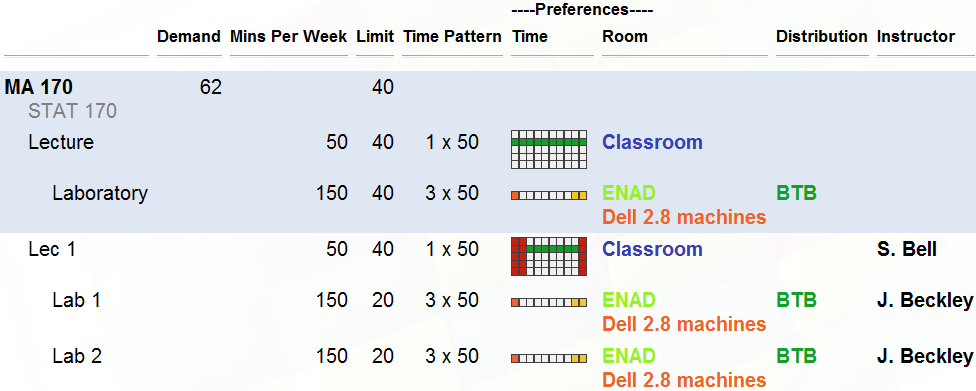
UniTime is an open-source course scheduling system designed for higher education institutions and still being updated and developed. It provides functions such as automatic class scheduling, classroom resource management, conflict detection, etc. The following analysis is based on its documentation and user feedback.

### **Analysis**



*User Interface provides an easy and intuitive means of data entry.*



*Preferences and requirements can be set on various levels of the course structure.*

*Classes are organized in a visual representation of the course structure. Preferences and requirements can be set at multiple levels.*

* **Automatic Class Scheduling**
  + The system can automatically generate a course schedule based on the availability of teachers, courses and classrooms.
  + **Pros:**
    - Support multiple scheduling rules (Teacher availability, classroom capacity).
    - Reduce manual operations and improve efficiency.
  + **Cons:**
    - Requires manual adjustment if there is a conflict.
    - No fixed time schedule for option.
* **Classroom Resource Management**
  + The system can manage class resources like room capacity, equipment configuration.
  + **Pros:**
    - Supports multiple classroom types, such as laboratories, multimedia classrooms.
    - Classrooms can be assigned automatically based on course subject.
  + **Cons:**
    - No real-time monitoring of classroom usage.
* **Conflict Detection**
  + The system can detect scheduling conflicts such as teacher time conflicts or duplicate room booking.
  + **Pros:**
    - Supports multiple rules such as exceeds room capacity.
    - Generate reports to help administrators resolve issues.
  + **Cons:**
    - Accuracy needs to be improved as sometimes there are false reports.
* **Data import and export**
  + The system supports importing data such as course information from other school systems and export scheduling results.
  + **Pros:**
    - Supports multiple file formats such as CSV, XML.
    - Can be integrated with other school management systems.
  + **Cons:**
    - Format incompatibility occurs sometimes.
    - Export content can not be customized.
* **User Interface**
  + The system provides a web-based user interface.
  + **Pros:**
    - THe interface is fully functional and supports a variety of operations such as schedule adjustment and conflict checking.
    - Support multiple language
  + **Cons:**
    - The overall design is relatively old-fashioned and the operation is not intuitive enough due to that.
    - Lack of support for mobile devices.

### **Conclusion:**

Through the analysis of Unitime’s documents, we can see that the system performs well in automated scheduling and flexibility, but has shortcomings in interface design, conflict detection and mobile support.

# **Requirements Analysis**

**FURPS+ Model:**

| **FURPS - MoSCow Analysis** | | | | |
| --- | --- | --- | --- | --- |
|  | **MoSCoW** | | | |
| **FURPS Requirements** | **M** | **S** | **C** | **W** |
| **Functional** | **Must** | **Should** | **Could** | **Wont** |
| Automatic Class scheduling | **√** |  |  |  |
| Attendance Management | **√** |  |  |  |
| Grade Management |  |  | **√** |  |
| Notification |  |  |  | **√** |
| Mobile Support |  |  |  | **√** |
| **Usability** | **Must** | **Should** | **Could** | **Wont** |
| Simple Interface | **√** |  |  |  |
| Intuitive Operation |  |  | **√** |  |
| **Reliability** | **Must** | **Should** | **Could** | **Wont** |
| Data Sync | **√** |  |  |  |
| Data Backup |  |  | **√** |  |
| **Performance** | **Must** | **Should** | **Could** | **Wont** |
| Quick Response |  | **√** |  |  |
| Efficiency |  | **√** |  |  |
| **Supportability** | **Must** | **Should** | **Could** | **Wont** |
| Feedback |  |  |  | **√** |
| Technical Support |  |  |  | **√** |

* **Functionality:**
  + **Automatic class scheduling function:** The system can automatically generate a class schedule and allow teachers to adjust it manually.
  + **Attendance management:** The system can automatically record student's attendance and generate reports.
  + **Grade Management:** The system can support different grading methods.
  + **Notification:** The system should send notifications about course changes and meeting reminders etc.
  + **Mobile support:** The system can provide mobile applications for teachers to view and operate at any time.
* **Usability:**
  + **Simple Interface:** The system interface needs to be clear and strict forward, easy to use.
  + **Intuitive operation:** The system’s operating procedures should be intuitive, easy to learn.
* **Reliability:**
  + **Data sync:** The system can ensure data synchronization between devices.
  + **Data Backup:** The system can back up data to prevent data loss.
* **Performance:**
  + **Quick Response:** The system should generate course schedules and process data quickly.
  + **Efficiency:** The system should be able to handle large amounts of data.
* **Supportability:**
  + **Feedback:** The system can provide feedback ways for suggestions.
  + **Technical support:** The system should have technical support to help teachers with the functions and features.

**MoSCoW Model:**

* **Must Have:**
  + **Automatic class scheduling function**
  + **Attendance management**
  + **Data sync**
  + **Simple Interface**
* **Should Have:**
  + **Quick Response**
  + **Efficiency**
* **Could Have:**
  + **Grade Management**
  + **Data Backup**
  + **Intuitive operation**
* **Wouldn’t Have:**
  + **Feedback**
  + **Mobile support**
  + **Notification**
  + **Technical Support**

# **Personas:**

**Persona 1: Department Head**

**Photo**: 

Source:https://media.istockphoto.com/id/494851564/photo/female-teacher-in-front-of-chalkboard.jpg?s=612x612&w=0&k=20&c=KViLqJN1bM8UvOyPSyx6lNGfYhM7Po2OYzuhoPckBc0=

**Name**: Dr. Huiyi Yu

**Role**: Head of CS department

**Goals**:

* Monitor staff performance and ensures there is no conflict with them
* Generate reports on grades, attendance, and documentation efficiently
* Make sure budgets and schedules are well managed
* Monitors faculty management system and make sure they integrate with the school system

**Pain Points:**

* Scheduling management may be difficult, especially when there is frequent conflict in schedules
* The current system doesn’t provide easy ways to track the students and staffs
* Gathering reports such as student history, grades, and class features can be inefficient and time consuming

**Brief Backstory:**

Since Dr. Yu is the head of the CS department, she is responsible for overseeing the aspects of the CS department operations. She has been the department head for 2 years.

**User Stories 1.1 :**

As the head of the department. I hope the system can automatically generate a class schedule. In order to reduce the time of manual scheduling and scheduling conflicts.

**Acceptance Criteria:**

* The system should automatically generate a class schedule based on the availability of teachers, courses and classrooms.
* The system should support manual adjustment of the schedule and provide conflict detection functions.
* The system should allow the setting of scheduling rules.

**User Stories 1.2:**

As the head of the department, I want the system to automatically generate grades, attendance and document reports.

**Acceptance Criteria:**

* The system should support the generation of student reports, attendance reports, and staff performance reports.
* The system should provide report templates to allow customization of report content.

**User Stories 1.3:**

As the head of the department, I hope the system can monitor class scheduling conflicts in real time. In order to resolve conflicts in a timely manner and ensure smooth scheduling of classes.

**Acceptance Criteria:**

* The system should detect conflicts like schedule conflicts, classroom overbooking and course overlap.
* The system should generate conflict reports and provide resolution suggestions.
* The system should allow manual adjustment of conflicting class schedules.

**User Stories 1.4:**

As the head of the department, I hope the teacher management system can be seamlessly integrated with the school system.

* **Acceptance Criteria:**  
  The system should support importing teacher information, course information and student information.
* The system should support exporting class schedules, grades and attendance data to the school system
* The system should provide logging of data import and export.

**Persona 2: Administrator**

**Photo: Source: https://thewarriormessenger.com/wp-content/uploads/2016/11/Liu\_1-900x826.jpg**

**Name:** Anson Liu

**Role:** College Administrator

**Goals:**

* Ensure data security and prevent scheduling conflicts
* Provide support to teachers for use of software and system
* Addresses user feedback from staff and students to improve system usability
* Handles faculty scheduling conflicts

**Pain points:**

* Software is incompatible for mobile devices
* Data backup and sync are slow and not reliable
* The system has a difficult interface, making difficult for some teachers to use

**Brief Backstory:**

Anson Liu manages the college’s schedules, system usability, and data management. He aims to make the faculties’ system reliable, efficient, and easy to use. He has worked as the college’s administrator for 7 years.

**User Stories 2.1:**

As a college administrator, I hope the system interface design is simple and intuitive. So that teachers and administrators can get started quickly and complete their work efficiently.

**Acceptance Criteria:**

* The system interface should be designed to be as simple as possible and avoid too many function buttons.
* The system should provide intuitive operation procedures to reduce learning costs.
* The system should provide help documents and tutorials to help users get started quickly.

**User Stories 2.2:**

As a college administrator, I hope the system can collect user feedback and improve based on the feedback to improve system availability and user satisfaction.

**Acceptance Criteria:**

* The system should provide feedback channels to allow teachers and students to submit feedback.
* The system should be improved based on user feedback and the improvement measures should be recorded.

**User stories 2.3:**

As a college administrator, I hope the system can support multiple grading methods such as weights of assignments and exams for teachers in order to flexibly calculate student scores based on course needs.

**Acceptance Criteria:**

* The system should support percentage, hierarchy and custom weighting.
* The system should automatically calculate a student's total score and generate a transcript.
* The system should allow teachers to manually adjust grades and record the reasons for the adjustments.

**User stories 2.4:**

As a college administrator, I hope the system can automatically generate the class schedule and allow me to adjust it manually.

**Acceptance Criteria:**

* The system should automatically generate the class schedule based on the availability of teachers, courses and classrooms.
* The system should allow teachers to manually adjust the class schedule and provide available time and classroom options.
* The system should detect scheduling conflicts and provide suggestions for resolution.

# **Requirements Validation**

| Requirements Traceability Matrix (RTM) | | | | |
| --- | --- | --- | --- | --- |
| Requirement ID | Description | Source | Priority | Status |
| F1 | The system should automatically generate a class schedule based on the availability of teachers, courses and classrooms. | Interview | Must | Drafted |
| F2 | The system should allow teachers to manually adjust the schedule and provide available time and classroom options. | User story 1.1 | Must | Drafted |
| F3 | The system should be able to detect conflicts and provide solution | User story 1.1 | Must | Drafted |
| F4 | The system should support multiple attendance methods | Interview | Could | Drafted |
| F5 | The system should automatically generate attendance reports and allow teachers to view and export them. | User story 1.2 | Could | Drafted |
| F6 | The system should support multiple scoring methods | User story 2.3 | Could | Drafted |
| NF7 | The system should provide a simple and intuitive interface | Interview | Must | Drafted |
| NF8 | The system should have efficient algorithm that handle data quickly | Interview | Should | Drafted |
| F9 | The system should have a notification function that notify teacher about their schedule | Interview | Wont | Drafted |
| F10 | The system should be able to import teacher information, course information and student information. | User story 1.4 | Could | Drafted |
| F11 | The system should allow manual adjustment of conflicting class schedules. | User story 1.3 | Should | Drafted |
| F12 | The system should provide mobile support | Interview | Wont | Drafted |