

Auto Scheduler

CS372 - Dr. Botting
Tuesday Thursday

By: Nieleh Wilcots, Raishon Montañez,
George Pangeti and Melissa Quintana

Auto Scheduler

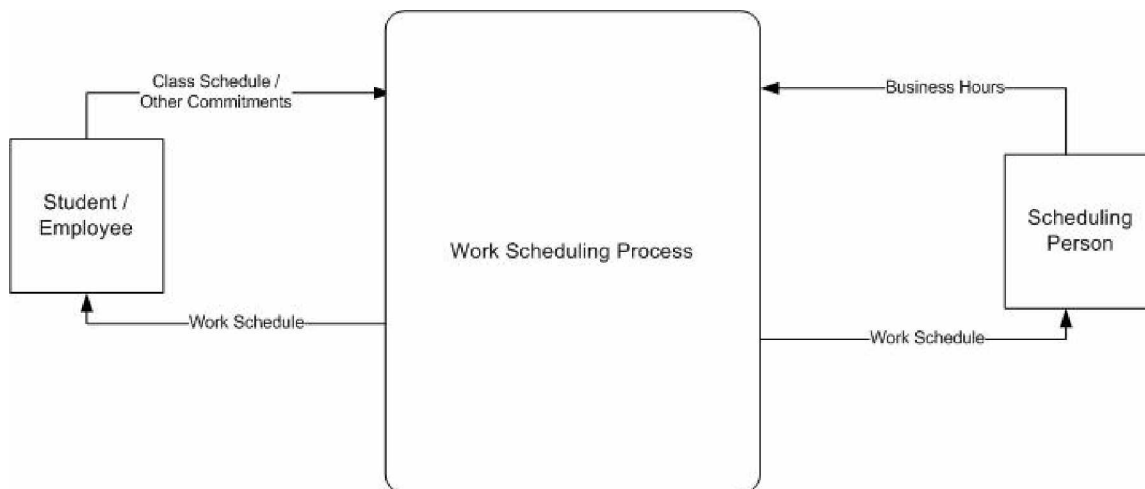
Nieleh Wilcots, Raishon Montañez, George Pangeti and Melissa Quintana

Executive Summary

Vision

Our goal is to ease operations in the department of Information Services at California State University of San Bernardino by freeing manual labor. The current manual system of computing student employees' work schedules would be replaced with an automated task. This allows the scheduling employee to be more productive by providing excess time to be delegated to other operations needed to be completed.

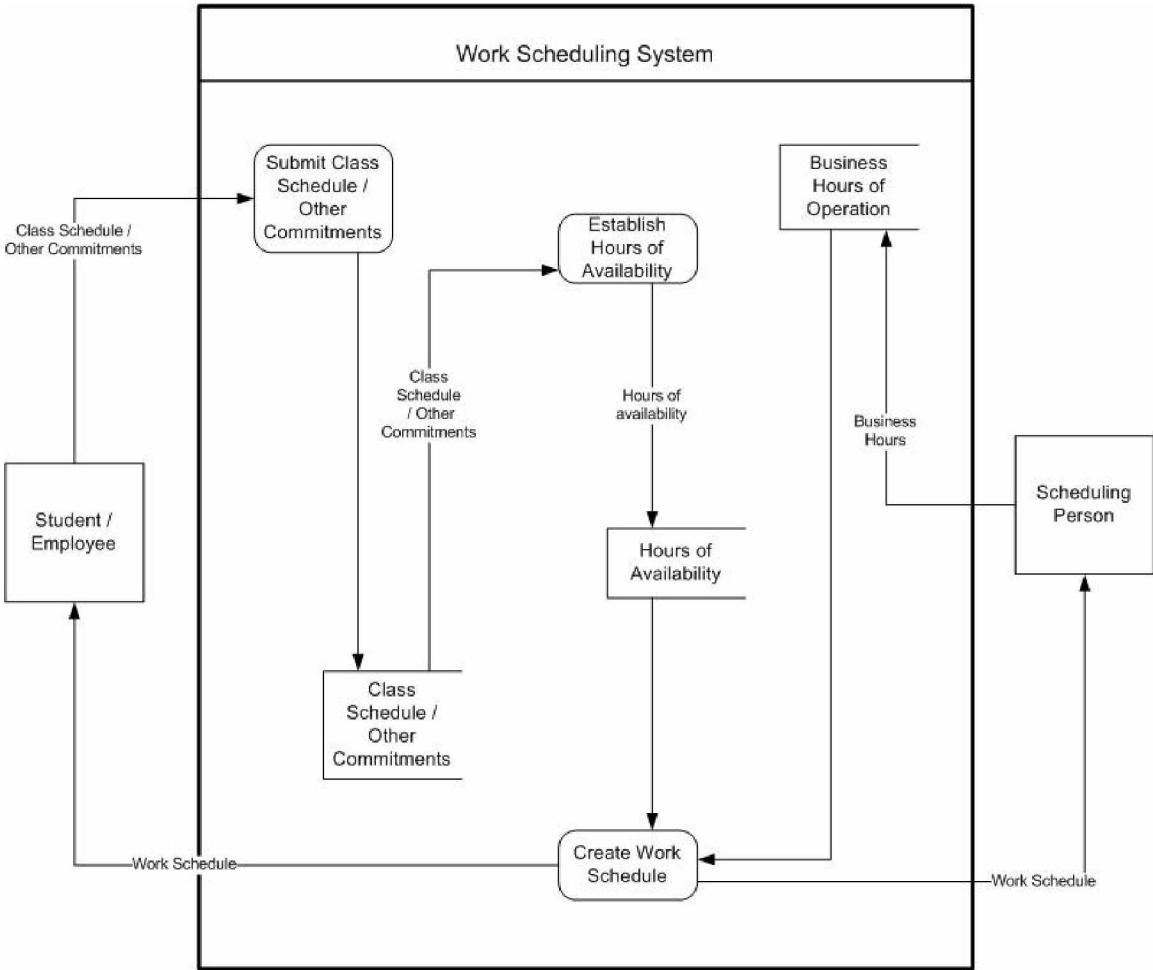
Context DFD



Auto Scheduler

Nieleh Wilcots, Raishon Montañez, George Pangeti and Melissa Quintana

Level 0/Fish Eye DFD



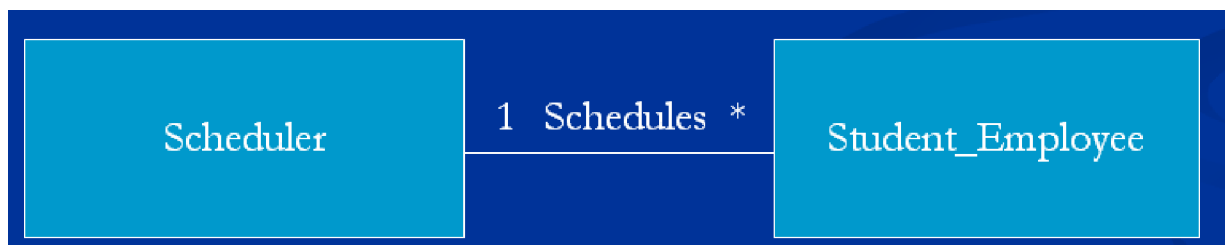
Auto Scheduler

Nieleh Wilcots, Raishon Montañez, George Pangeti and Melissa Quintana

Data Base

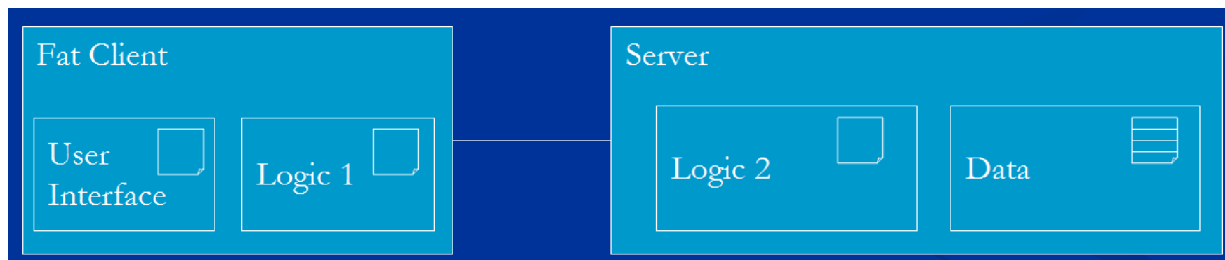
- Department (**DeptCode, DeptName)
- Course (**course #, time, day, dept, title)
- Student_Employee (**Student #, **employee #, Name)
- Schedule (**employee #, Name, Schedule)

Entity Relationship diagram -- -- --(ERD)



- The Scheduler can have many Student_Employees
- the Student_Employee will only exist in the scheduler once.

Architecture

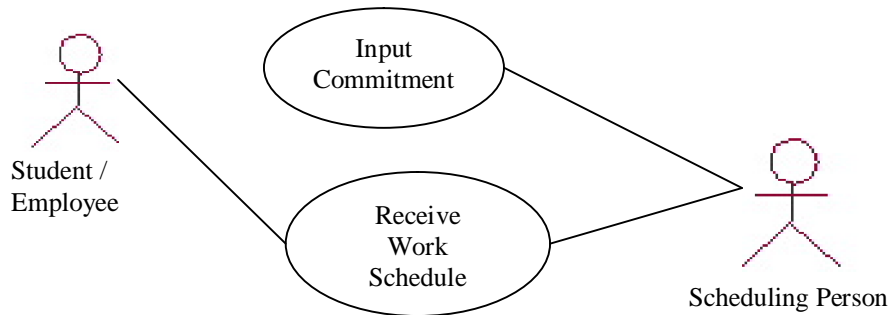


- We will use a Fat client architecture. The client will perform the bulk of the data processing operations. The data itself will be stored on the server.

Auto Scheduler

Nieleh Wilcots, Raishon Montañez, George Pangeti and Melissa Quintana

Use Case Diagram



Use Case Scenario

Name	Create Work Schedule
Actors	Scheduling Person, Student/Employee
Stakeholders	Student Employees
Main Success Scenario	1. Scheduling Personnel inputs employees other commitments 2. System calculates work schedule 3. System prints out work schedule
Alternative	After step 3, work schedule has collisions* 1. Scheduling Person makes manual changes

*collisions: student work schedules overlapping, or not filling in all business hours with student employees.

Auto Scheduler

Nieleh Wilcots, Raishon Montañez, George Pangeti and Melissa Quintana

Non-Functional Constraints

- Security

Lock machine to prevent hacking from within the organization

Prevent unauthorized tampering with schedules.

Prevent leaking of staff information from database

- Interfaces with other systems

If database is down or not working (no functionality)

Auto Scheduler

Nieleh Wilcots, Raishon Montañez, George Pangeti and Melissa Quintana

Implementation Plans

Development Plan

The sequence in which we plan to develop and implement the new system starts with the implementation of the database in using mysql. Following the database will be the implementation of the software. The software will reference and manipulate the data within the database for our desired output. And lastly, we will create the user interface using QT. There is only one use case so the exact order of development is exactly as it was just described.

Start Up

Once the development plan is complete, we will introduce the new system with the Big Bang method. By this we mean that we plan to immediately implement the new running system to replace the old way of manual scheduling. Although, we are taking this method of low cost and high risk, the manual operations will still be available if needed.

Data Conversion

The initial data is currently on hard copy do to the fact that the task currently is a manual operation. The vital part of this process is collecting all of the important information to be inserted into the new system and eliminating all of the unnecessary information, which in turn will provide simplicity.

Training

All training will be face to face. When someone is given the assignment of scheduling, they will be given thorough, hands on, face-to-face training. Currently, George (one of the creators of the new system) is the only one in the information systems department who does the scheduling.

Testing

The testing being provided from the start of the system to the systems end of life will be Acceptance training. This is a great advantage to us considering George is going to have consistent use of the system while it is being built and after it is built. He will be the user who specifies and carries out the tests.

Maintenance

We also have the advantage of continuous corrective maintenance. From the start of the project to the end of the projects life we will be able to detect any bug or defects and fix them.

Auto Scheduler

Nieleh Wilcots, Raishon Montañez, George Pangeti and Melissa Quintana

Cost Benefit Analysis: Project Auto Scheduler

Implementing this project will computerize a tedious manual job, freeing up time for staff to focus on more complicated issues.

1. Cost:..... Programming Software.
2. Tangible or intangible: Intangible
3. Once or recurring: 10 weeks
4. How much: \$0
5. When does it happen? Project Start

1. Cost:..... Transferring software to tangible medium.
2. Tangible or intangible: Tangible.
3. Once or recurring: Once
4. How much: \$2 - \$5
5. When does it happen? Project End

1. Cost:..... Training.
2. Tangible or intangible: Intangible
3. Once or recurring: 1 week (1 hour a day)
4. How much: \$50 (at \$10 an hour)
5. When does it happen? Upon implementation of software

1. Benefit: Save valuable time.
2. Tangible or intangible: Intangible
3. Once or recurring: Approximately one week every quarter
4. How much: \$200
5. When does it happen? Project Start