Joseph Thornton

Bianca Aguirre

Daniel Bui

Justin Langshaw

Sameeh Khan

Hasib Habibi

CSC 170

SRS document

Department Professor Schedule cards

**Contents**

[Purpose](#_30j0zll) 3

[Document conventions](#_1fob9te) 3

[Project scope](#_3znysh7) 5

[References](#_2et92p0) 5

[**Overall description**](#_tyjcwt) **5**

[Product perspective](#_97thuksa8is4) 5

[User classes and characteristics](#_1t3h5sf) 6

[Operating environment](#_4d34og8) 6

[Design and implementation constraints](#_17dp8vu) 7

[Assumptions and Dependencies](#_td9ly2387g8l) 7

[**System features**](#_dxfk8hi1kfwh) **7**

[System feature x](#_wumqq5zt5on) 8

[Description](#_35j2rb3qjjg4) 8

[Functional requirements](#_hhv6tm1d8xjc) 9

[**Data Requirements**](#_hng0g4ct8yze) **9**

[Logical data model](#_q9oxidu6v5be) 9

[Data dictionary](#_mb68x2we4snd) 10

[Reports](#_1hzst7hzetik) 11

[Data acquisition, integrity , retention, and disposal](#_k1zycu6ykc46) 12

[**External interface requirements**](#_4i7ojhp) **13**

[User interfaces](#_riloiaqq8as1) 13

[Software interfaces](#_cgblw2qclo9w) 13

[Hardware interfaces](#_nxxn59bgv6o) 14

[Communications interfaces](#_oybca4wur282) 14

[**Quality attributes**](#_qsh70q) **15**

[Usability](#_306pnjz3zde1) 15

[Performance](#_fx9b5zr5w8x3) 15

[Security](#_vm4rn9v2dc2h) 15

[Safety](#_dybh50egq6e) 15

[[others]](#_147n2zr) 16

[**Internationalization and localization requirements**](#_3buftbwh4s7v) **16**

[Other requirements](#_23ckvvd) 16

[**Appendix A: Glossary**](#_xvd5i5sr1x3e) **16**

[**Appendix B: analysis Models**](#_e8hmd9heup84) **16**

Introduction

## Purpose

The product to be made is information cards for the Computer Science Department to place on the outside of the office of each professor at the beginning of the semester. The purpose of this product is to supply the students of the Computer Science department with the information of each professor. The information provided will include the department, the semester, the Professor's name, their office room and hours as well as the classes they are teaching. Which will include their courses, the sections, days and time as well as the room. This document is created and intended for the development team. The team will consist of the developers, project managers, marketing staff, testers, and documentation writers. This SRS document will be used as a tool for the development team to use while creating the product.

## Document conventions

**i. Print Format**





**iii. User GUI view teacher look up**

## Project scope

A database to hold the faculty information card.

1. The information card will have the faculty member which teaches specific courses, there will be different sections of one course that will be held on specific days, times and rooms.
2. It will also include the office room number to the professor, the phone number and also the office hours of date and time.
3. The cards can be printed in one operation and also individually.

UI of the database that will allow for information to be withdrawn from it using queries. This will allow for easy updating from database to a readable document.

The database will be monitored and maintained regularly.

With this new system, the office administrator will not have to generate new cards each semester because the system will gather information from previous semesters.

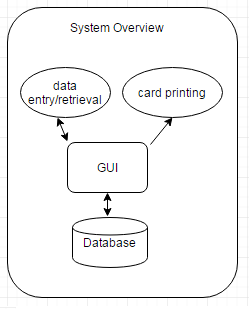
Also, the office administrator will have sole access to changing faculty members while the clerk will only be able to faculty office hours.

## References

# Overall description

## Product perspective

This product originated from the need to encapsulate faculty schedules into easily accessible physical cards that can be used by faculty members and students alike. This will be one standalone system that can be accessed by an administrator and a clerk only. It will consist of a database to hold the faculty information, a GUI that can allow entry and retrieval of various faculty members and classes, a once a semester printing operation that will consecutively print all faculty member cards, along with an option for printing individual faculty member cards on request.



## User classes and characteristics

## Operating environment

User Interface and Platforms

* Windows
  + Executable Application for managing back end database directly (manager option)
* Android
  + Android app for managing database directly (manager option)
  + Visual side app for viewing and printing cards
* Apple Products
  + Iphone/IOS application for managing database directly (manager option)
  + Visual side app for viewing and printing cards
* Web Application (HTML)
  + Main user side for PC visual for viewing and printing cards
* Apache tomcat
  + Used for hosting web app for PC

Location

* SQL database will be located on CSUS campus servers hosted in Riverside hall
* Users can be mobile but must have a secure VPN account to connect
* Tomcat server will be located on titan and run constantly (may require java update)
* (Second to above tomcat server is run off campus and uses database on campus)

## Design and implementation constraints

## Assumptions and Dependencies

Since the stakeholders are going to be Computer Science professors, it is reasonable to assume that they all will be adept at utilizing the database. The database will be dependent upon updates provided by the CSC department. The faculty will be providing yearly updates, then the database can be fulfilled by the professors.

# System features

This system must implement all of the following features that are not deemed optional. it must have printable faculty cards organized by viewing gui for printing convenience. some kind of search engine used to find teachers,classes,department etc. the system will allow for multiple search and find with GUI to send data to printing table after completed. (optional) It will return a list of students currently under teachers supervision along with perspective grades. the system will also have a web accessible portion for viewing the cards before they are printed. this feature will allow teachers to see cards that are coming out to ensure correctness if necessary. the administrator will have data side manipulation and entry such as account creation new hires and more discussed later. (optional) The admin must have the ability to export cards into different doc types in order to cross platform barriers. All information from previous semester will be brought in with the option for the teachers to update their information so the admin does not need to start from scratch each semester.

**terminal breakdown:**



## System feature x

1. view cards
2. search and assemble teachers
3. change and update database
4. send specified data to a card
5. concatenate cards

## Description

1. The card view will be just like the visual inside of the layout section of this document it will allow the user (any user) to view cards at their discretion if a user finds a problem or wants to change a part of a card they must have the right privileges but if they do they may alter and change cards as they please
2. Each person will have the ability to search the database through the web gui provided to all the users. this will allow the user to bring together any set of data for a department or sub section and create a card ticket this ticket will be stored and mabey viewed by others or changed depending on privileges.
3. The admin will have the authority alone to insert teachers and their respective details manually however the admin will be able to send a secure link to teachers if he wishes it to be done externally and the teacher may alter the database according to the specific need but not further.
4. This feature will allow the user to send a card to be viewed or printed after they have found the information needed and selected proper elements they can press send just like in the visual above and the card will be ready for printing or viewing.
5. any user can take multiple cards and create a new concatenated version of it for example if there are cards for CSC and CPE departments they can simply create a new card and select concatenate.

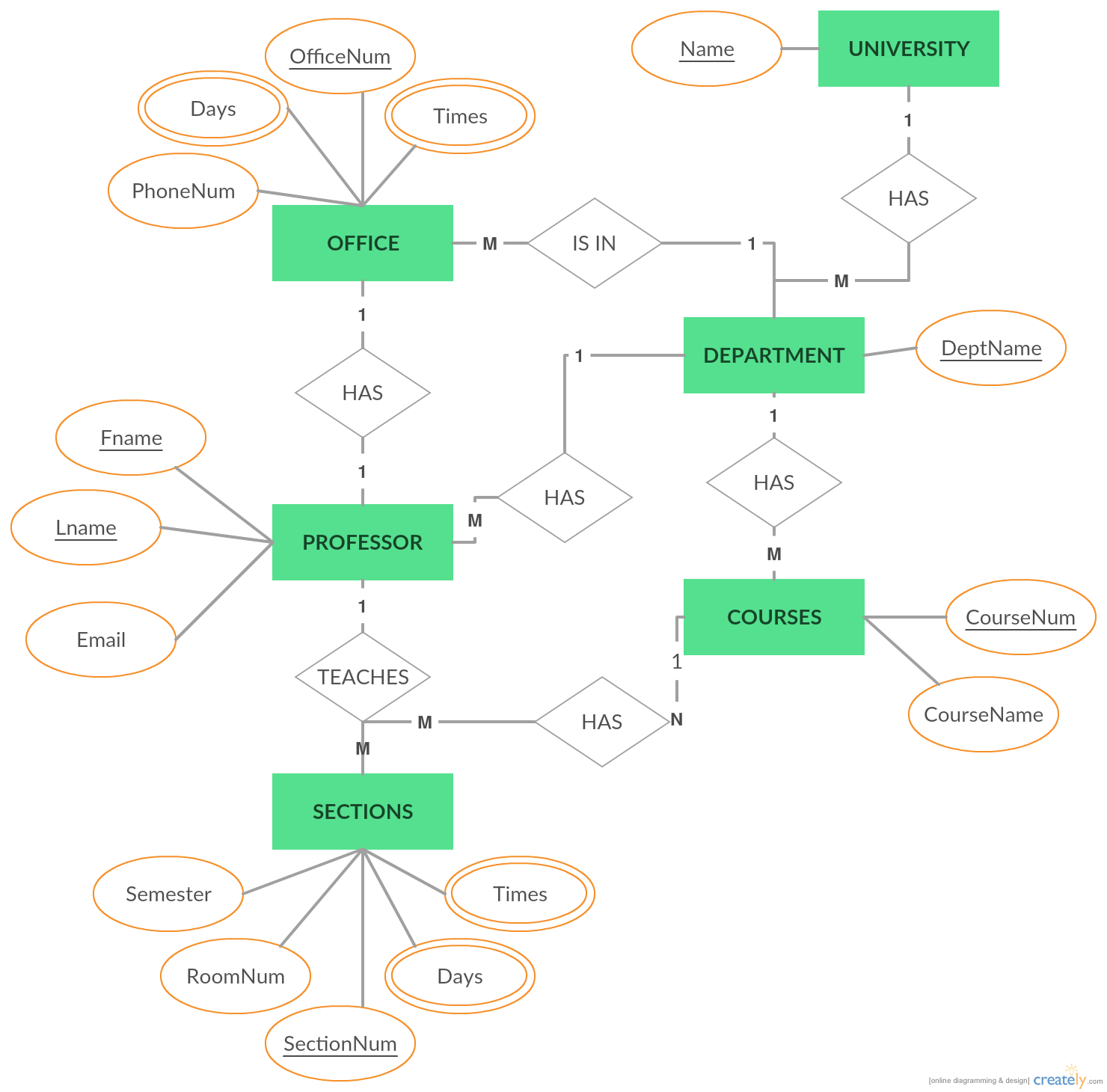
## Functional requirements

* 1. the view must be resizable for multi platform use it must scale with the window that it is given
  2. the view will only allow for viewing of one card at a time (for coding reasons) so if select a card i cannot tab to another card i must return first
  3. it must show the data in the organized fashion shown in the above sections
  4. drag and drop may be implemented if time provides (for organization purposes.
  5. the database must be accessible to all authorized users for search purposes only! the primary keys will be teacher names and departments with the option of SSN if provided

1. the admin must have access to change entries in sql database in order to maintain a current state. all other users can search but cannot modify the database.
2. the user must be able to first use A and then create a card by sending the card to the database this will allow other users to view and/or alter it
3. multiple card listings are located on the main screen of the whichever GUI you are using the user must be able to create a card and the select concatenate and view this screen in order to select cards to join

# Data Requirements

## Logical data model

Entity Relationship Diagram of Database  


## Data dictionary

Name: Has

Description: Has stands for the relationship between course, department's, office, and sections. This means that is is included in theses as features or has them as a reference through a foreign key.

Name: Teaches

Description: This describes the relationship between professor and sections. This shows that the teacher is teaching these sections. Provides information about the section they teach.

Name: Is In

Description: this is the relationship between office and department. This states that the office is located within the department.

## Reports

Only report that will be generated is a Confirmation Report:

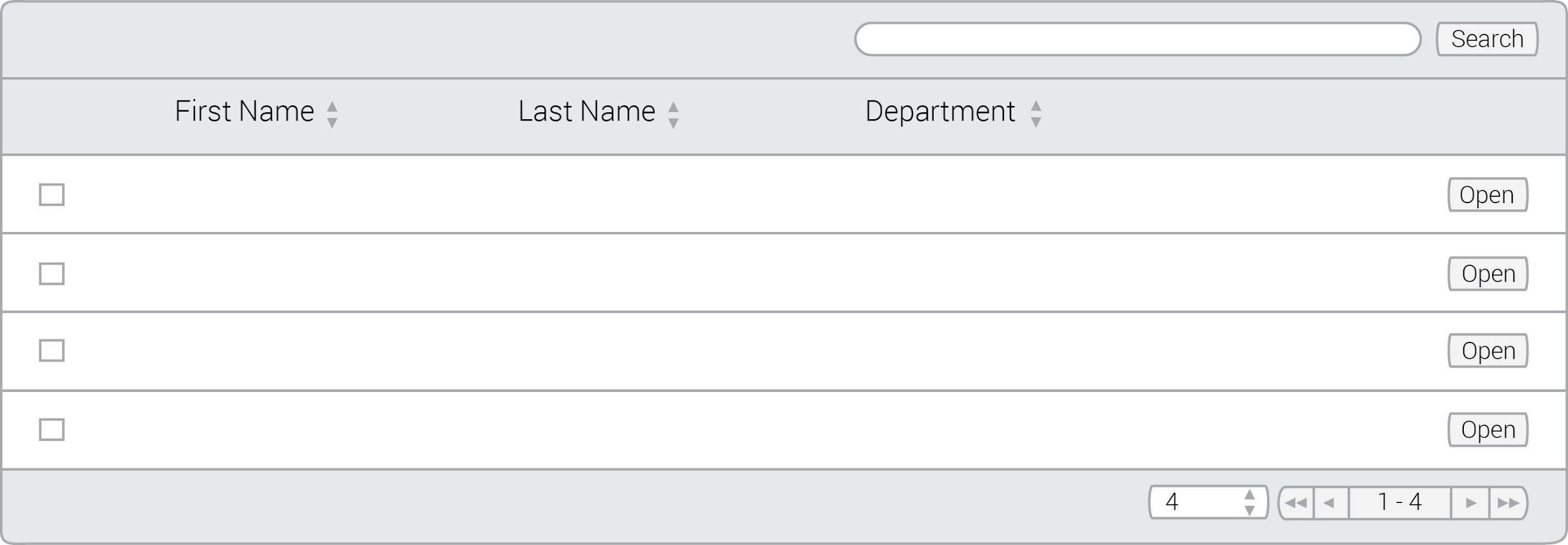
|  |  |
| --- | --- |
| Report Title | Confirmation Report |
| Report Purpose | To show confirmation of printed faculty cards |
| Decisions Made From Report | To see if all faculty members have their cards printed for them. |
| Priority | 1 |
| Report Users | Main Printing Application |
| Data Sources | Main Printing Application, Faculty Information Database |
| Frequency and Disposition | · Report is static  · The report is generated every time the cards are printed  · all of the faculty member fields are accessed, only one transaction is needed  · printing the cards trigger the report  · the report is generated automatically  · the report will be received by the faculty administrator. It will be printed and sent to them by email. |
| Latency | The report must be delivered by email as soon as the cards finish printing. |
| Visual Layout | · Portrait  · Size 8 ½ by 11 inches  · Report includes no graphs |
| Header and Footer | · Printing Complete  · Page x of y  · Time of printing  · Name of Clerk who printed  · Data pulled from Faculty Database  · CSUS: Computer Science Department  · Copyright CSUS |
| Report Body | · Faculty members that have cards printed:  · Fields included: Faculty Members printed for  · Parameters: Faculty fields  · Single column, each row a faculty member successfully printed for  · N-1th row: any faculty members that couldn’t have their cards printed  · Nth row: any errors reported while printing  · Header will be in Calibri (Body) format, 12 point font, bold.  · Each body field will be in Calibri (Body) format, 11 point font.  · Error row will be same as above but in bold.  · Text field overflows will go over into next row.  · No calculations or transformations are shown.  · Faculty members are sorted in alphabetical order. |
| End-Of-Report Indicator | End of report will say “End of Report” |
| Interactivity | The report is not interactive |
| Security Access Restrictions | Only Admin and Clerk can see this report. |

## Data acquisition, integrity , retention, and disposal

To confirm the accuracy of the database’s contents it would be imperative to consult the hours and class times with the Sacramento State catalog. For updating the database, the system would need to transfer over any recurring data that would come between semesters. One would need to perform backups, checkpointing, and mirroring to assure integrity.

# External interface requirements

## User interfaces

The user interface will be a web application that the Department can use to view and print out the information cards. The interface will be very accessible and straightforward. We will use the Bootstrap framework to have a simple and clean white and gray theme with san serif web fonts. The screen will be responsive so it can be viewed on all different sized screens. The main page will display a table list with all professors and their information. There will be a filter box at the top of the screen to select individual professors, departments, courses and sections. You will be able to click on any professor’s name and it will take you to a different page with their information card. This page will then allow you to edit and or print the individual cards. We want the user interface to be as straightforward and easily accessible as possible for the departments to use. 

## Software interfaces

JDBC: the jdbc will provide sql connection for both the stand alone app and the web app to connect , android app will have to use a different API not yet determined and will not be present in product upon release same for Apple support except via website.

Java servlets: the servlets will contain the HTML code for the website it will have the basic html structure with margins and spacing set the feal will be implemented using css code. with the added access to functionality in the utility functions.

Java Servlet utility functions: these will be the logic provided to the servlets for processing such data as usernames and passwords. and any input fields that are send out.

Android studio: again this will not be implemented on first release, further knowledge of android studio and subsequent api’s will be needed

Java GUI interface: this will be coded in java swing for simplicity. this can be altered in the future. this allows for back end management to remain the same reducing the amount of time the project will take.

apple GUI: further knowledge of apple API’s will be needed here aswell but both android devices and apple devices will be able to use the web application.

all the software services will share the same sql database. the java gui and website will use the same API the JDBC the others will use their own respective API’s.

System Software Layout:



## Hardware interfaces

Describe the characteristics of each interface between the software components and hardware components, if any, of the system. This description might include the supported device types, the data and control interactions between the software and the hardware, and the communication protocols to be used. For our program no hardware interfaces have been identified.

## Communications interfaces

The program will utilize and network connection that will allow to connect to a database. The connection will be an HTTP connection through the network to an interface that displays the database. The interface will then be able to be pulled up through the network onto an mobile device or desktop for accessing the database.

# Quality attributes

## Usability

* Ease of Learning attributes include:
  + Verbose prompts
  + Wizards to guide setups
  + Visible Menu options, such as printing option
  + Meaningful, plain-language messages
  + Help screens and tooltips
  + Similarity to other familiar systems
  + Limited number of options and widgets displayed
* Ease of use attributes include:
  + Keyboard shortcuts
  + Auto completion of entries
  + Auto correction of errors
  + The system must be capable of printing all of the cards in one operation for the initial posting. It must also be capable of printing an individual card on request.
  + Ease of use metrics include:
    - Time for printing confirmation no more than 4 seconds
    - Time for data to be sent to database with confirmation no more than 2 seconds
    - Clerk must be able to enter 95% of data without needing any help
    - Data entry must be designed to allow 10% chance of error while filling out
    - Number of clicks to get to any type of task/information should be 1-3
    - All 1 page data entry should have a confirmation click before submittal

## Performance

There are specific performance requirements for various system operations. If different functional requirements or features have different performance requirements, it’s appropriate to specify those performance goals right with the corresponding functional requirements, rather than collecting them during the process.

## Security

Regarding security and privacy issues, our product intends to not include any when being used in the computer science department. The cards will only include information that’s approved of from the department and faculty members.

## Safety

this product have few safety requirements but there are a few user situations that must be avoided.

1. admin passwords must remain in the knowledge of the administrator alone the password must not be distributed to the staff under any circumstances. Some information in the database may be private and not open to the public.
2. the database logic is to be managed by this team alone any exterior changes will result in maintenance fee to the customer!

## [others]

# Internationalization and localization requirements

Since this product will only be used across the California State University of Sacramento, we will not need any internationalization and localization requirements. Instead, we will have the requirement that it will be valid across all colleges and departments in Sacramento State. Although this system is created for the Engineering and Computer Science Department, it is still applicable for any other department on campus. Since each department has all the same requirements for Professors, their courses, sections and office hours, it will be easy to stretch this product across campus. Each department will be able to have access to the software and input their own data on each professor into the system.

## Other requirements

# Appendix A: Glossary

GUI : Graphical User Interface

CSUS : California State University: Sacramento

VPN : Virtual Private Network

# Appendix B: analysis Models

System Overview: Page 6

System Features: Page 7

Logical Data Model: Page 9

System Software Layout: page 12