Deliverable 3
"Process Modeling" (D3)

Class Voting System
IS436 Structured Systems Analysis and Design
10/25/2019

Arian Eidizadeh - Project manager

Cyril Fonzock - Software analyst

Michael Woldgerima - Developer

Leul Assamenew - System analyst

Sanee Patel - Data analyst

Yohannes Feleke - Developer

1) Develop process models using Visio, **Lucid Charts** (or any other drawing tool you are familiar with). The process models will include the context and level diagrams. After drawing your Level 0 diagram, detail each process in your level 0 diagram as much as possible. Therefore, you can go until level-n for each process at level 0, where n is the level at which all aspects of the business process are explained precisely. Please closely follow the notation given in PowerPoint. Your DFDs should be readable from top-left to bottom-right and should avoid crossing lines as much as possible. Make sure that the DFDs that you produce are decomposed in a meaningful way and use the principles of balancing. Do not violate the rules of data flow diagramming.
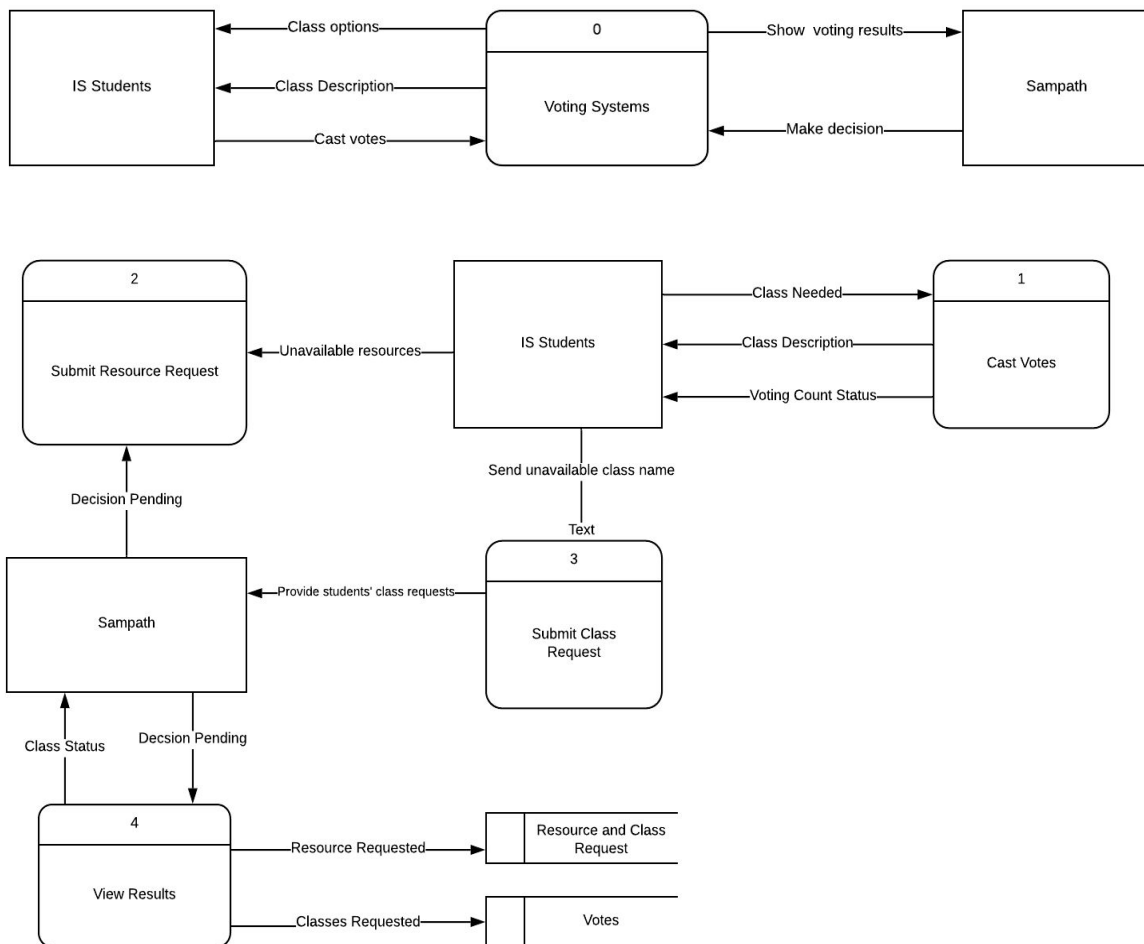
2) Revisit your Deliverable 2. Change the requirements definition and use cases that you prepared according to what you have learned by going into more detail in DFD design.

Functional Requirements
1. **Learn more about the software.**

1.1.   The voting system itself should be an ease-of-use tool for students to vote for their 2 most desired class.

1.2.   Process-oriented design will be used as it can allow the system to check incoming and transfers to vote as well efficiently.

1.3.   The system should allow students to view degree requirements and course list while voting for classes.

2.   **Implementing the software.**

2.1.   Student will log into the website using their CAMPUS ID. The website can be accessed through a web browser on any device.

2.2.   The student has two votes they can use and click on whatever class they would like to take.

2.3.   Once their ID is verified by comparing it to a valid student database, the voter can complete their ballot and securely submit their vote(s) to the web application ballot box.

3.   **Logic behind the Software.**

3.1.   In terms of our keyword analysis machine learning development- we will use ML to scan submitted sentences, put them into keywords, and push them out in a nice-clean format for the director.

3.2.   We will use a supervised learning, predictive module- neural based algorithm for this assignment. 80% Training and 20% test data in the alpha stage

Non functional

In the nonfunctional requirement, decisions are made about the user interface, the hardware and software, and the system's underlying architecture. When it comes to nonfunctional requirements for this project,

1.   **Operational**

1.1.   Voting system will be a web application that can be accessed on any device (it will have to have the correct configurations).

1.2.   Voting System should have the updated classes for the next year.

1.3.   Compatibility with browsers on desktop, mobile, and laptops/etc should all be tested and working.

2.   **Performance**

2.1.   Interactions between the user and the system should not exceed 2 to 3 seconds.

2.2.   The system downloads new status parameters **instantly**. Program should be making changes in real **time**.

2.3.   The system should be available for use 24 hours per day, and however long the department director keeps the voting system online.

2.4.   The system supports 250-300 simultaneous users at a time. **If there is a need for more resources, we will allocate them.**

3. **Security**

3.1.   Dr. Sampath will be the only one with direct access to the voting results. Users will not be able to see other text submissions.

3.2.   Developers will test the software for potential bugs.

3.3.   The system will provide the proper mitigations against XXE, SQL, XSS, and other OWASP Top 10 vulnerabilities.

4. **Cultural and Political**

4.1.   Students' vote information is protected in compliance with Data protection.

4.2.   Votes administered into the system will only be submitted from students. Teachers, facility and others will not be allowed to vote.

4.3.   The department director (Dr. Sampath) will authorize the start and end date of the voting system.

Use Case #1 : Cast Votes   Priority: High
Actor: UMBC Information Systems Students
Description: This use case describes how UMBC Information Systems Students will be able to cast two votes for two classes.
Trigger: When UMBC IS student clicks on class "Cast Vote" Button will be made visible
Type: External Trigger
Preconditions:
1. UMBC Information Systems Students are authenticated
2. Class Polling System is available and online
3. Classes are available and online

Normal Course:
1.0. Cast vote for classes
   1. Login is authenticated $\Rightarrow$ CampusID
   2. Retrieval of classes is initiated $\Leftarrow$ Class options
   3. Classes are displayed or not (see Alternative Course 1.1) $\Leftarrow$ Class details

4. Class selection redirects you to class description ⇒ ClassID

5. Student cast vote on class description page ⇒ Vote

6. Return to step 3 for second vote

7. After second vote student, page redirected to Use Case 3

Alternative Courses:

1.1. IS classes fail to be displayed

    1. System display error is displayed ⇒ ErrorID

    2. Students submit notification of error ⇒ Select error

    3. Exit Use Case

    4. Sampath is notified of error ⇐ Error notification

    5. Sampath notifies system support team ⇒ System support notice

PostConditions:

1. Student votes are submitted and approved
2. Sampath is notified for display error

Use Case #2 : View Results   Priority: High

Actor: Sreedevi Sampath

Description: This use case describes how Sreedevi Sampath will be able to view the results of the polls when the deadline is reached.

Trigger: When system deadline is reached results will be collected.

Type: Temporal Trigger

Preconditions:

1. Sampath is authenticated
2. Deadline is reached
3. Results are available and viewable

Normal Course:

1.0. View Results

    1. Login is authenticated ⇒ AdminLogin

    2. Results are retrieved and displayed ⇐ Result view

2.0 Allow Results to be viewed.

    1. Results are sent in whatever format is presented and sent by Dr.Sampath

Alternative Courses:

1.1. Result are unviewable

    1. Sampath informs System support team of failure to view results   ⇒ Systems support notice

    2. System support team attempts to fix error   ⇒ ErrorID

Postconditions:
1. Sampath uses the results to for the upcoming year classes
2. Sampath requests System support team to fix the problem

Use Case #3: Resource and Class Request (Machine Learning)   Priority: High
Actor: UMBC Information Systems Students
Description: This use case describes how UMBC Information System Students will be able to propose classes or resources that they believe would be useful as an IS Major.
Trigger: When students finish voting
Type: External Trigger
Preconditions:
1. Students are authenticated
2. Students cast their votes

Normal Course:
1.0. Submit Resource and Class Request

    1. Student inputs in message box   ⇒ InputID

    2. Student submits proposal   ⇒ Submit proposal

    3. Request is sent and stored in a database   ⇒ Request stored in DB

Alternative Courses:
1.1 Resource and Class Request is not viewable

    1. System support team is informed of the failure to view results   ⇒ Systems support notice

    2. System support team attempts to fix the error   ⇒ ErrorID

Postconditions:
1. Requests are used for predictive modeling
2. Sampath looks through requests

Lessons Learned:

- We have tested and conducted more research into what ML algorithm we will use and where the process will be set during our DFD design

○ We will use a supervised predictive neural based algorithm that is designed to recognize patterns
● We have changed how quickly our database and website update from 8 minutes to it working in real time.
● Added a use case normal course for Dr.Sampath having the ability to send results to students
● Edited various pieces of information within our functional and nonfunctional requirements.

3) Provide a simple text definition for each process, entity, datastore and each data flow in your diagram. Pay attention to the data flow names – each should be unique if they are carrying unique data.

● **Class options**: All of the classes a user can take are displayed for the user to choose from and respond too accordingly. The voting system should display these classes correctly and every class should fit the schedules for the following year.
● **Class description**: Each class should have its own description that the voting system displays. The voting system should describe the professor teaching the course, the location, textbook requirements, pre-requisites and a brief summary of what is to be taught.
● **Cast votes**: This is the main function for selecting the necessary courses for the coming year. Students should be capable of selecting 2 classes for the upcoming academic year. A vote will mark with a check mark or x to signify their vote has been selected and counted in the system.
● **Show voting results**: Dr.Sampath should be able to view the results of the class votes. She will use this information to make knowledgeable decisions about what classes need more professors, what classes can be cut and what classes can remain the same. As a result students will be able to take all of their necessary classes towards graduation.
● **Make decision**: This is where the professor will select what classes will be made available to the students the following year. Results will either be mailed to students or/and made available to students that took the survey. This way students will know if the results they picked made an impact on what classes they wanted to take.
● **Decision Pending**: Sampath lets the system and user know that she is in the process of making her decision. It can't be rushed because the necessary data needs to be inputted first. The decision will be pending until a final decision is made.
● **Class Status**: Shows the status of what classes are currently being selected by students and the current results. Dr.Sampath is who looks over the data so she would be the only one to view this. If any changes are needed to be made she can start to make early deliberations.
● **Voting Count Status:** The count of all votes is made visible to the students to view. This is important as it lets the students know what their chances are for getting professors to support their class. If they can't get the support they might ask their friends to fill out the survey or simply become more informed of the status of their class.

- **Resource Requested:** Makes a request for more resources whether it be more professors, programs or anything of the like. Budgets, materials, and lab utilizations will decide how much resources will be provided.
- **Classes Requested:** Displays what classes are requested by students for the next academic year. These classes should be displayed by the system so that both sampath and students alike can view the information.

4) Include your updated project plan.

Project Purpose:

We plan to build a voting system that will provide more insight into what classes need prioritization ahead of time (One Year).

Reasoning:

Throughout various semesters in the IS department, different classes become more popular and become unavailable quickly. This means that some students will be waitlisted and may never be able to take that class, forcing them to take courses that they are not interested in. In most cases, the head of the IS department will have to find another staff member to teach the popular class many students request. The issue with this is that this process is time-consuming and may finalize before classes start, disrupting many schedules that were already pre-built.

We will continue to discuss progress and conduct daily meetings to observe and fix any issues we may face.

High-Level-overview
- The website greets student, asks them to enter campus ID
  - Authenticates student ID
- Website will then introduce the student to the multiple classes that are going to be offered (One year before)
- The student has two votes they can use
  - They click on whatever class they would like to take
- After a week, the voting systems goes offline.
- The voting system will take the results and output them in various ways
  - Graphs, Charts,
- The voting system will track all votes to see if there is a trend over the years (can use ML or just data analytics software)

- The website will allow students to enter any input for what they'd like to add to the voting system
  - Such as adding specific programs to computer labs (MS Project, etc)
  - Machine learning to scan keywords/the most common inputs to save time for the director of information Systems. Neural-based predictive model.
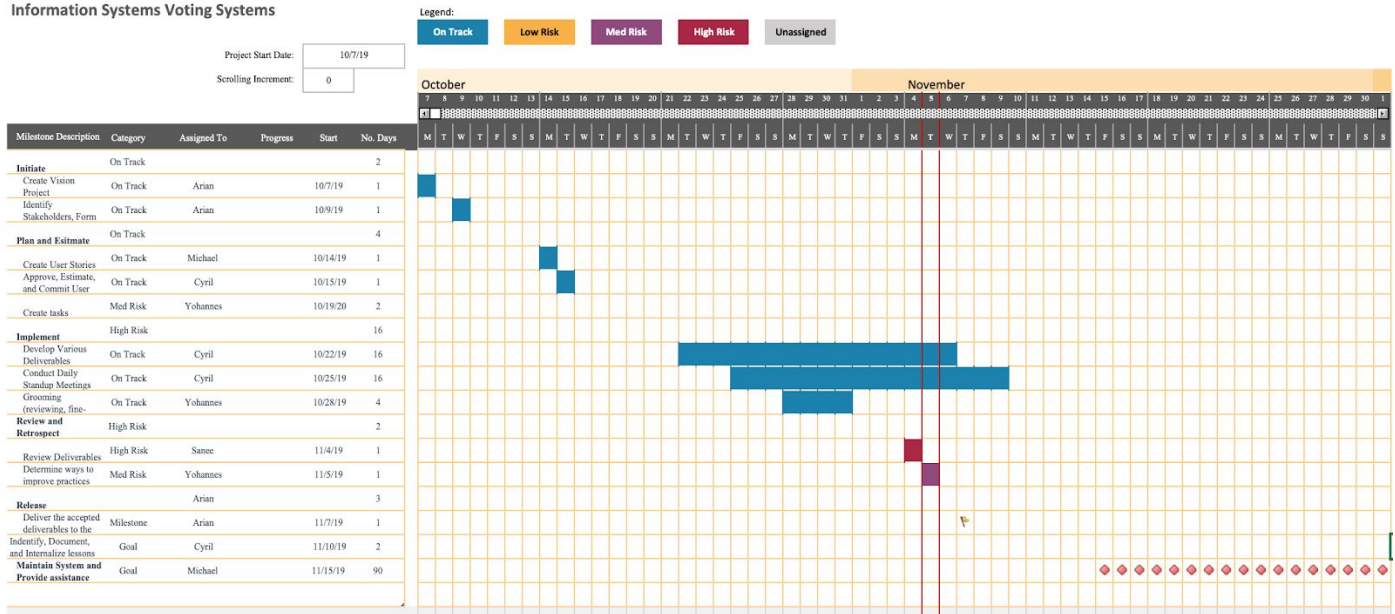
Above is our Project Work Plan updated with on Task/Completed tasks (Blue). The Project work plan includes responsibility for each team member along with what the risk-level, milestone, and goals for the future are.

Our next tasks include (Reviewing Deliverables (Medium Risk) and Delivering our deliverables (High Risk as we need to ensure all systems are operational).

5) Create a Kanban board and assign the requirements to your team members.