Deliverable 2
Requirements Definition Document and Use Cases

Class Voting System
IS436 Structured Systems Analysis and Design

Arian Eidizadeh - Project manager
Cyril Fonzock - Software analyst
Michael Woldgerima - Developer
Leul Assamenew - System analyst
Sanee Patel - Data analyst
Yohannes Feleke - Developer

Interview

- The name of our interviewee is Dr. Sampath,
  - An Associate Professor and the Director of Undergraduate Programs for the Information Systems Department.
- The interview was conducted on September 23
- The interviewer, Arian Eidizadeh, first pitched the project idea to Dr. Sampath and told her the project purpose and the overall project scope.

Interview Continued:

Example Interview Questions:

- Before we begin various project planning phases, would this project be feasible for the Information Systems department?
- When developing this project what restrictions would we have to consider?
- How does the Information Systems Department plan ahead in terms of assigning teachers to classes?
- Is the proposed budget a reasonable price?
- What do you believe would be beneficial to implement within the voting software
- Will this save time and resources during the class planning phase?

Colleague Questionnaire:

These questions were distributed by our group mates to various colleagues in the span of a week from 10/2/2019 - 10/9/2019. We asked them to answer whichever questions they believed had the utmost importance.

1) As a student, would you implement this voting system as well?

- **In total: 75% of students said they would implement this.**
- **There was a medium-high response rate**

2) Do you think your friends would implement this voting system?

- **62.5% of Individuals who responded to the question stated their friends would implement this system.**
- **There was a medium response rate to this.**

Functional Requirements

1. Learn more about the software.

   1.1. When it comes to the functional aspect of this project, this software should be able to let students vote on upcoming classes to see what they appeal to the most.

   1.2. We will be using process-oriented when it comes to the functional aspect of this voting system because it allows the system to check incoming and transfers to vote as well.

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

2. Implementing the software.

   2.1 The voter would need to log into a website/software we develop on a device. From there, the voter would submit their CAMPUS ID to authenticates

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

   2.3 Once their identity is verified, the voter can complete their ballot and securely submit their vote(s) to the web application ballot box.

Functional Requirements

3. Logic behind the Software.

3.1 Our machine learning will scan those sentences, put them into keywords, and push them out in a nice-clean format for the director.

3.2 That’s why we chose to use Predictive Modeling as our artificial intelligence technique it effectively identifies key attributes that a customer has entered before and determine actions that the customer is most likely do.

3.3 This allows us to safeguard from any errors we might occur when dealing with customer information. After a week, the voting goes offline and only Dr.Sampath looks over the votes.

Non Functional

1. **Operational**
   1.1. The voting system is a web application. So you can operate it on tablet, laptop, apple or android devices.
   1.2. The voting system should be able to interact with schedule of classes which are available now.
   1.3. The system should be compatible with any web browser such as Chrome, Firefox, Internet Explorer.
2. **Performance**
   2.1. Any interaction between the user and the system should not exceed 2 to 3 seconds.
   2.2. The system downloads new status parameters within 8 minutes of a change.
   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 simultaneous users at all times.

Non Functional Continued

3. **Security**

   3.1 Only director Dr. Sampath can see the voting results.

   3.2 Developers are able to work on maintenance work and make changes when errors occurred.

   3.3 The system includes all available safeguards from viruses, worms, and Trojan horses etc.

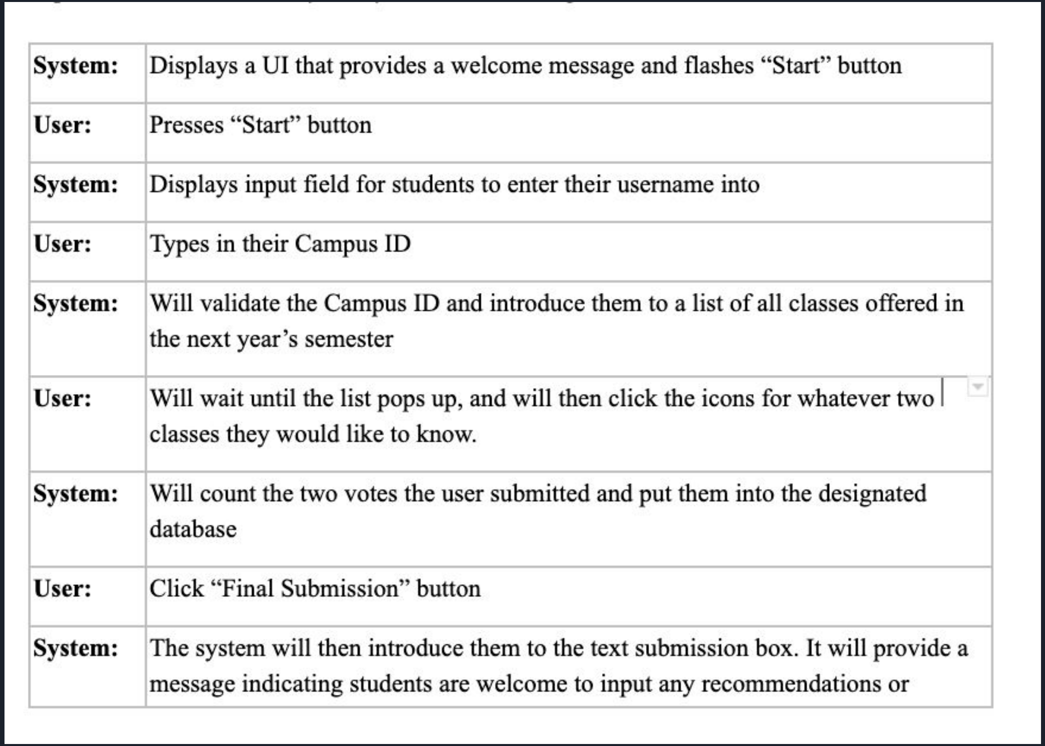
4. **Cultural and Political**

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

   4.2 The voting system policy is to get voted from only students, not from faculties.

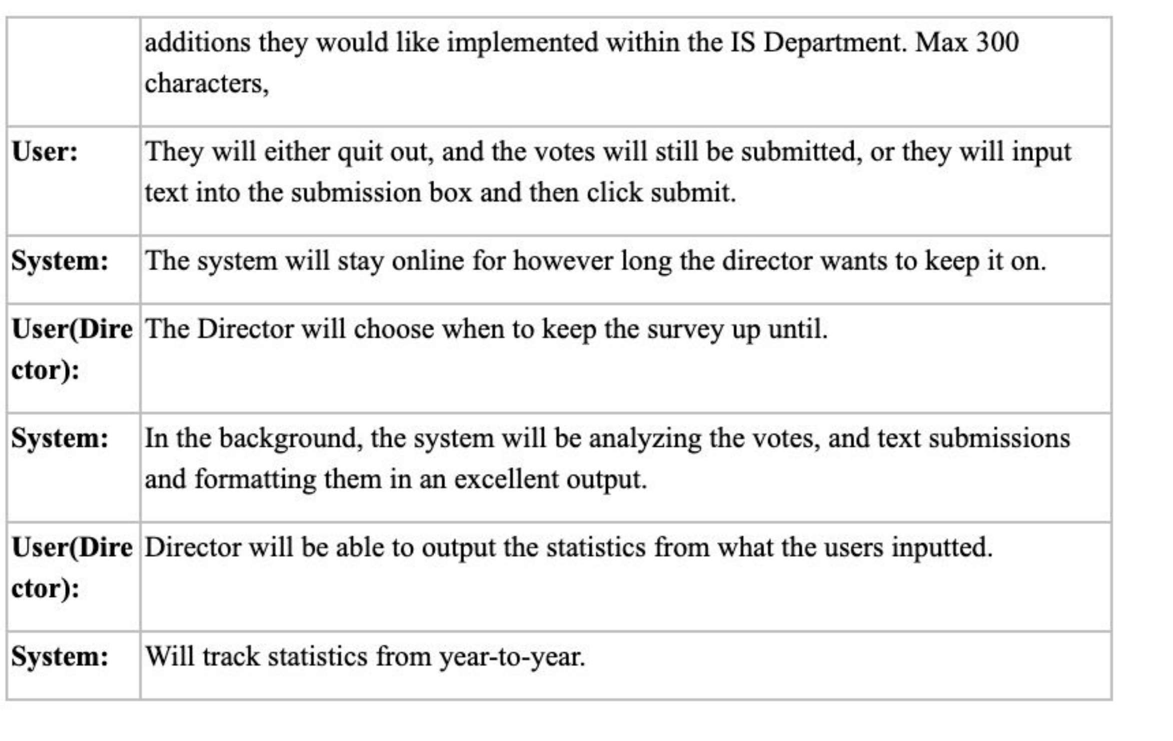
   4.3 The department director (Dr. Sampath) will authorize students and send them emails when she put voting system online.

Observation Notes



System:	Displays a UI that provides a welcome message and flashes “Start” button
User:	Presses “Start” button
System:	Displays input field for students to enter their username into
User:	Types in their Campus ID
System:	Will validate the Campus ID and introduce them to a list of all classes offered in the next year’s semester
User:	Will wait until the list pops up, and will then click the icons for whatever two classes they would like to know.
System:	Will count the two votes the user submitted and put them into the designated database
User:	Click “Final Submission” button
System:	The system will then introduce them to the text submission box. It will provide a message indicating students are welcome to input any recommendations or

Observation Notes Continued



	additions they would like implemented within the IS Department. Max 300 characters,
User:	They will either quit out, and the votes will still be submitted, or they will input text into the submission box and then click submit.
System:	The system will stay online for however long the director wants to keep it on.
User(Director):	The Director will choose when to keep the survey up until.
System:	In the background, the system will be analyzing the votes, and text submissions and formatting them in an excellent output.
User(Director):	Director will be able to output the statistics from what the users inputted.
System:	Will track statistics from year-to-year.

Document Analysis

- The first document is organization chart with the different teams and their respective objectives and goals. Every team has their respective managers and the project manager oversees all of the groups as a whole.
- The first form is a UML diagram of the website. It describes how the website will work as a whole and what different functions the website will have.
- The second form is a diagram of the first to third normal form of the database as well as database diagrams. Database fills requirements of First-Third Normal form.
- The final form is a registration form. This form will have all of the necessary documentation in regards to classes that will be offered and what necessary signatures were taken to allow the system to be in place.

Use Case Analysis and Use Case Documents

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

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

Use Case Analysis and Use Case Documents (Cont’d)

Use Case #3: Resource and Class Request (Machine Learning) Priority: High

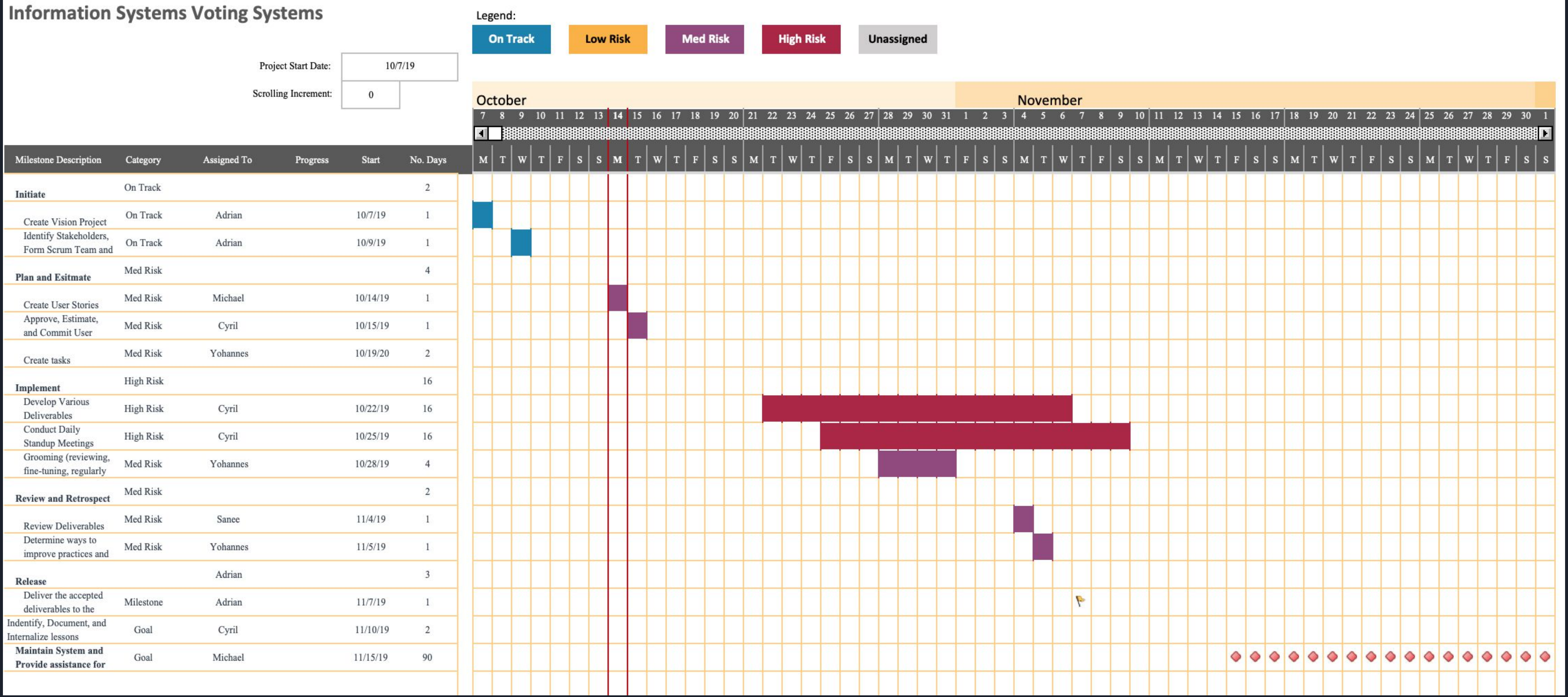
Actor: UMBC Information Systems Students

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

Excel Tracking Sheet



Information Systems Voting Systems

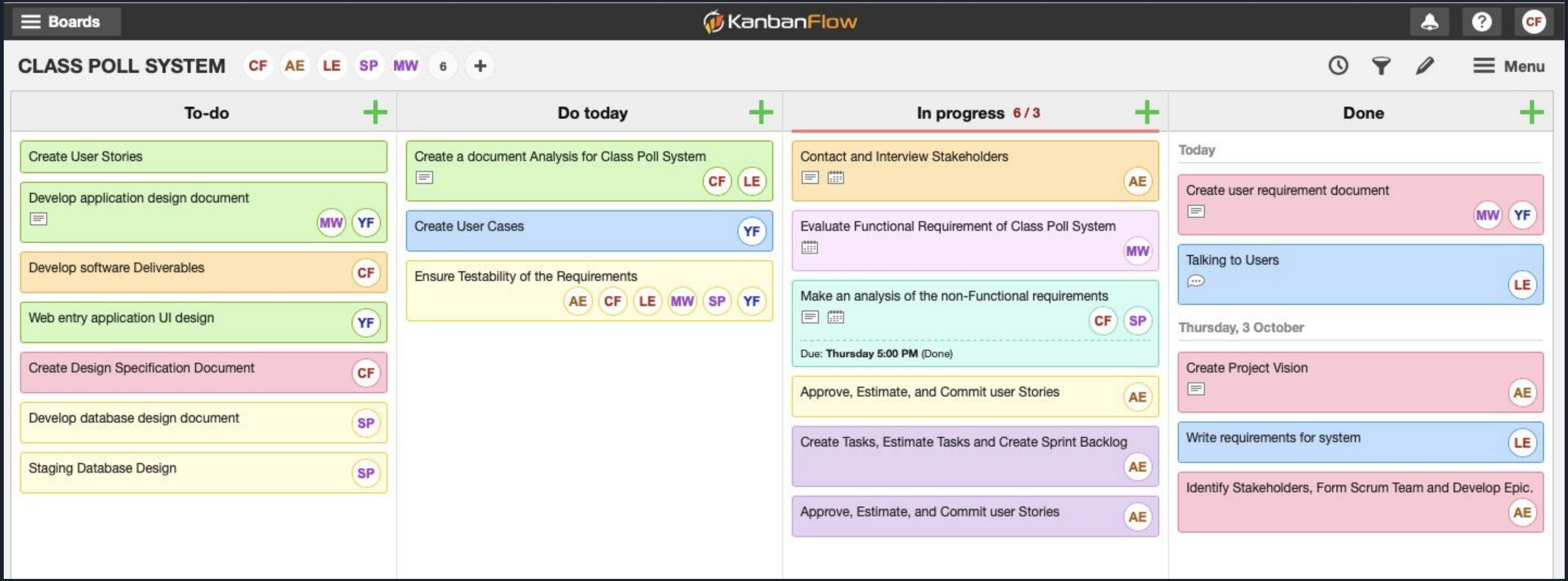
Project Start Date: 10/7/19

Scrolling Increment: 0

Legend: On Track | Low Risk | Med Risk | High Risk | Unassigned

Milestone Description	Category	Assigned To	Progress	Start	No. Days
Initiate	On Track				2
Create Vision Project	On Track	Adrian		10/7/19	1
Identify Stakeholders, Form Scrum Team and	On Track	Adrian		10/9/19	1
Plan and Esitmate	Med Risk				4
Create User Stories	Med Risk	Michael		10/14/19	1
Approve, Estimate, and Commit User	Med Risk	Cyril		10/15/19	1
Create tasks	Med Risk	Yohannes		10/19/20	2
Implement	High Risk				16
Develop Various Deliverables	High Risk	Cyril		10/22/19	16
Conduct Daily Standup Meetings	High Risk	Cyril		10/25/19	16
Grooming (reviewing, fine-tuning, regularly	Med Risk	Yohannes		10/28/19	4
Review and Retrospect	Med Risk				2
Review Deliverables	Med Risk	Sanee		11/4/19	1
Determine ways to improve practices and	Med Risk	Yohannes		11/5/19	1
Release		Adrian			3
Deliver the accepted deliverables to the	Milestone	Adrian		11/7/19	1
Indentify, Document, and Internalize lessons	Goal	Cyril		11/10/19	2
Maintain System and Provide assistance for	Goal	Michael		11/15/19	90

Kanbanflow



CLASS POLL SYSTEM CF AE LE SP MW 6 +

To-do

- Create User Stories
- Develop application design document MW YF
- Develop software Deliverables CF
- Web entry application UI design YF
- Create Design Specification Document CF
- Develop database design document SP
- Staging Database Design SP

Do today

- Create a document Analysis for Class Poll System CF LE
- Create User Cases YF
- Ensure Testability of the Requirements AE CF LE MW SP YF

In progress 6 / 3

- Contact and Interview Stakeholders AE
- Evaluate Functional Requirement of Class Poll System MW
- Make an analysis of the non-Functional requirements CF SP
  Due: **Thursday 5:00 PM** (Done)
- Approve, Estimate, and Commit user Stories AE
- Create Tasks, Estimate Tasks and Create Sprint Backlog AE
- Approve, Estimate, and Commit user Stories AE

Done

Today

- Create user requirement document MW YF
- Talking to Users LE

Thursday, 3 October

- Create Project Vision AE
- Write requirements for system LE
- Identify Stakeholders, Form Scrum Team and Develop Epic. AE