Relational Models

Sunday, October 16, 2016 4:15 PM

This table will store all the area of studies available.
Example: cs, Computer Science
StudyField

- Id
- Name

This table will store the possible Programs stores the field
And the type such as Bachelors, Minor etc.
Example: cs, Bachelors
Program

- Id
- FieldID
- Type (Check in (Bachelors, Minor))

FieldID References StudyField(Id)
Unique (FieldID, Type)

This table will store all the possible requirements for
All majors
Example: Mathematics, 6
Requirement

- Id
- name
- NumClasses

This table will store the requirements for each and all programs. From this
Should be able to grab all the requirements for a program.
ProgramRequirements

- ProgramId
- RequirementID

ProgramId References Program(Id)
RequirementId references Requirement(Id)
Unique(FieldID, RequirementID)

This table will store the user's base info such as email address password etc
(Will use the devise for this)
User

- UserId
- FirstName
- LastName
- …
- Password

This table will store users with their prospective programs.
This is so that a user can have more than one program
UserProgram

- Id
- userId
- programId

unique (userId, ProgramId)

This table will store requirements and their respective User Programs
This will tell us this requirement is for this user and for this program through programID

UserProgramRequirements

- id
- RequirementID
- UserProgramId

Unique(requirementId, UserProgramId)
FOREIGN KEY RequirementId REFERENCES Requirement(

This table will map a class to a specific requirement for a program for a user
This is so that users can user the same class for two or more requirements

ClassRequirement

- id
- UserProgramRequirementsId
- UserClassId

Unique(RequirementId, UserClassId)

Table will store the classes that each user takes

UserClass

- Id
- CourseNumber
- LetterGrade
- StartDate
- EndDate

FOREIGN KEY CourseNumber REFERENCES OfferedClass.CourseNumber

This table is a readonly table that stores all the Classes offered from which the user can choose from

OfferedClass

- CourseNumber
- name

This table will be used to store Components such as quizzes, labs etc for all users

Component

- id
- UserClassId
- name
- weight
- numAssigDropped (default 0)

FOREIGN KEY UserClassId REFERENCES UserClass.Id

This table will be used to store all assignments for all users

Assignment

- id
- componentId
- Name
- score
- maxscore

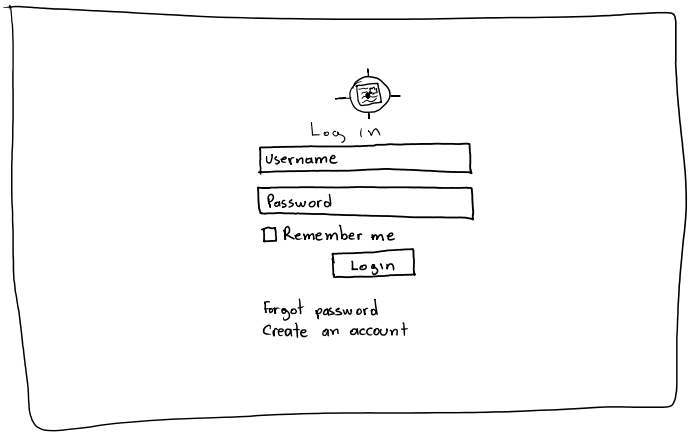
FOREIGN KEY componentId REFERENCES Component.Id

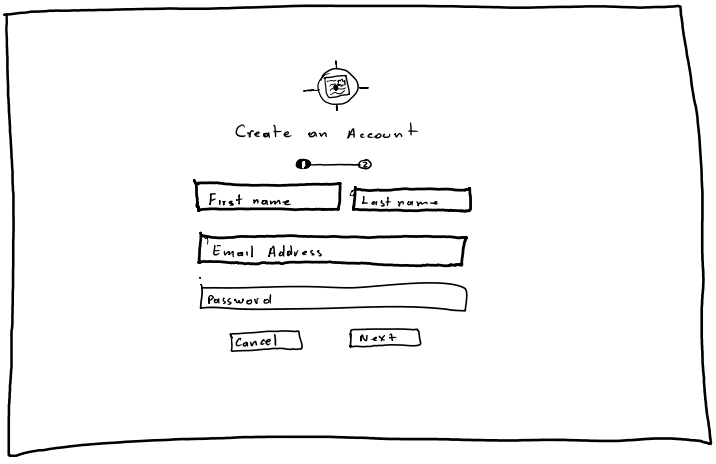
Controllers

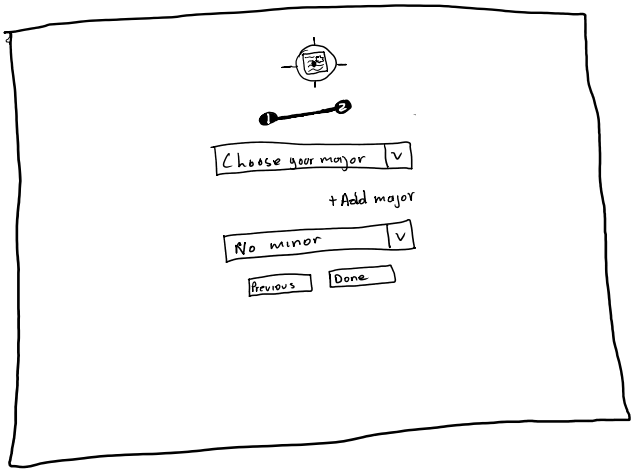
Sunday, October 16, 2016 4:22 PM

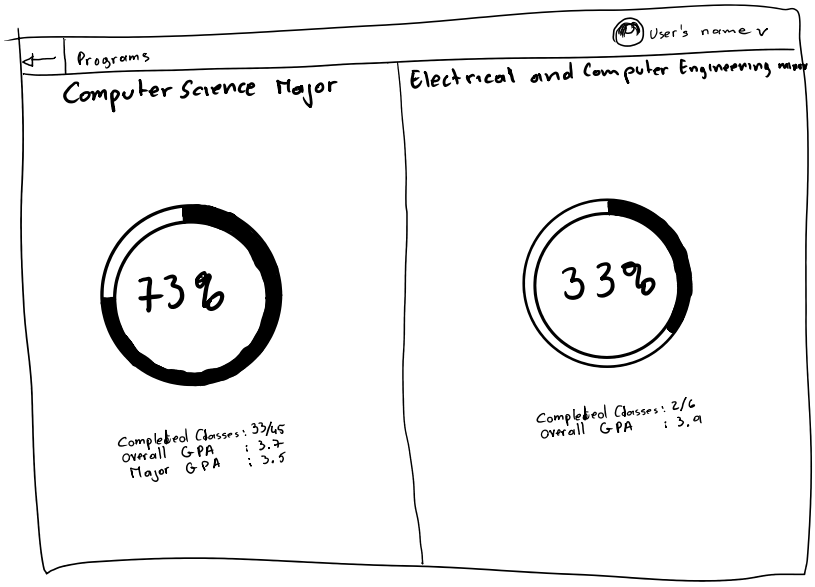
Rough UI Design

Wednesday, May 4, 2016 12:59 PM

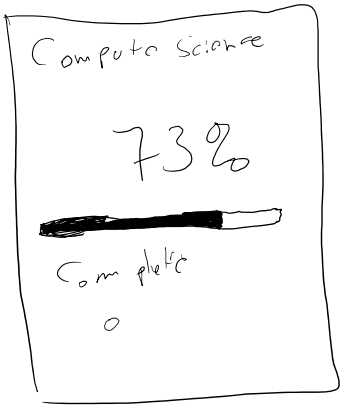


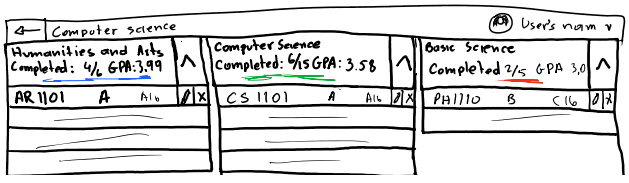


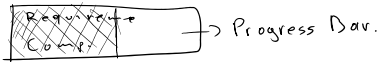


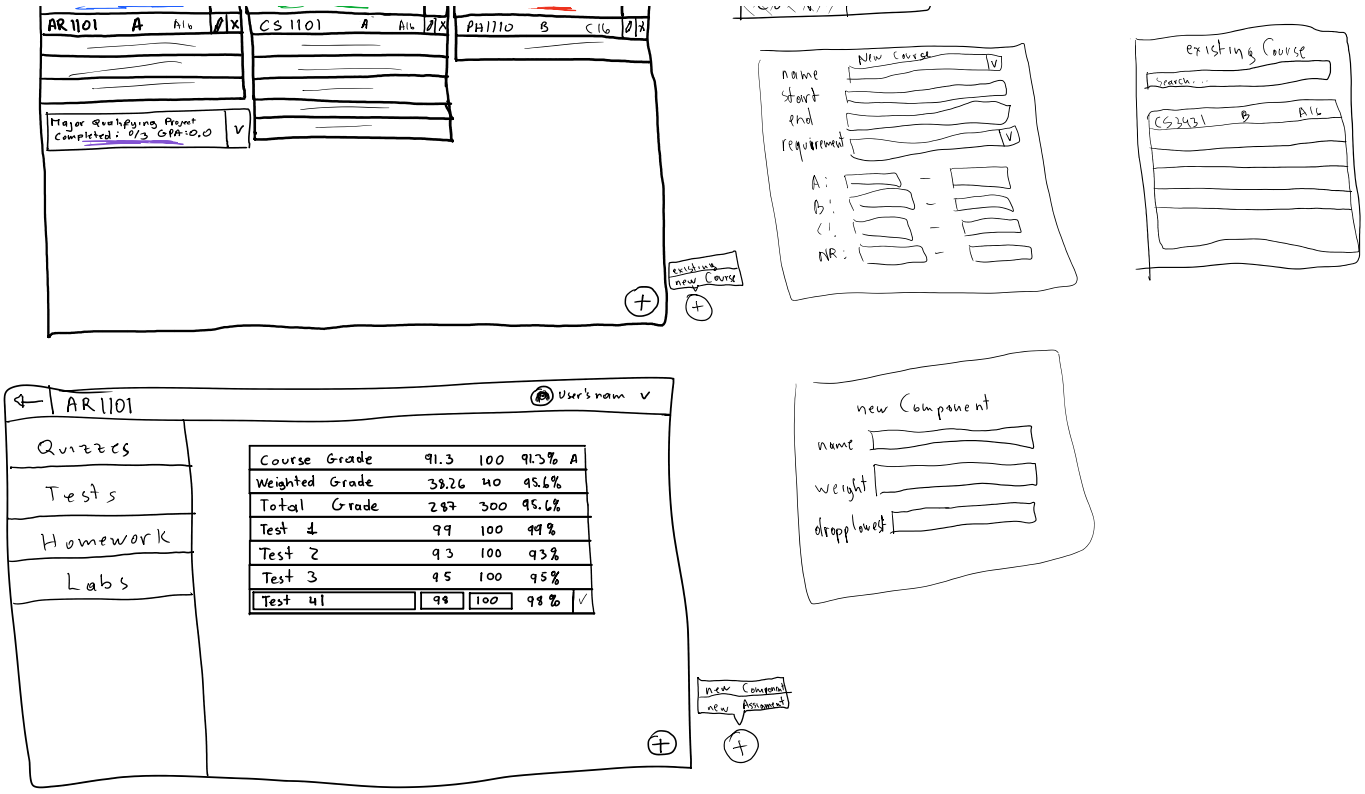


OR








AR 1101 A A16
CS 1101 A A16
PH 1710 B C16
Major Qualifying Proj: Completed: 0/3 GPA: 0.0
existing
new Course
New Course
name
start
end
requirement
A:
B:
C:
NR:
existing Course
Search...
CS 3431 B A16

AR 1101
User's nam
Quizzes
Tests
Homework
Labs
Course Grade 91.3 100 91.3% A
Weighted Grade 38.26 40 95.6%
Total Grade 287 300 95.6%
Test 1 99 100 99%
Test 2 93 100 93%
Test 3 95 100 95%
Test 41 98 100 98%
new Component
new Assignment
new Component
name
weight
drop lowest