

## CanvasPath

**CanvasPath** is a startup project which aims to help Lion State University maintain their software system for course information management. John Smith, a stakeholder of the project, along with members of the Lion State Advisory Board have been examining the current course management system and realized that there is a serious need to migrate the legacy system into one with a new design and up-to-date technology. The Advisory Board of Lion State (which luckily has some members graduated from Penn State University) unanimously identified **Canvas** as the primary reference for improving their system. Furthermore, to gain insights from Canvas users, the board suggest Mr. Smith to invite EECS students of Penn State University, who are known for their creativity and robust technical skills, to be involved in the project.

Mr. Smith, who is not a technical person, reached out to Dr. Lee and his team for assistance to carry out a feasibility study. After extensive discussions, they agreed that this study with guidance from Dr. Lee and his teaching team, can be performed by having students in CMPSC431W to systematically design and implement prototypes for validation of feasibility. The goal is not to produce a complete implementation of the system, as it would require too much time and resources for the students. Instead, students will focus on i) the process of the database design that may pose a significant risk due to uncertainty and the lack of understanding in requirements, and ii) the prototyping of certain system functionality as a proof of concept. It is anticipated that based on carefully examination on the tested prototypes, valuable insights and lessons may be obtained. Indeed, this design-prototype approach of feasibility study is often adopted when the requirements are not well understood. Moreover, a successful prototype can potentially serve as the foundation for the future production system in operation, and an impressive demonstration of the prototyped system, functions, and unique features will attract appreciation of the PSU talents by the members of the LSU Advisory Board who are mostly executives in S&P 500 companies.

Dr. Smith and his associates have tried to communicate this project abstractly and expects students to figure out and fill in all the missing details. Basically, you (students in CMPSC431W) will design a database-backed web application to manage course-related information, including courses, students, faculty members, and more. Students (you) will also implement a prototype to demonstrate system functions and your design. As parts of the prototype, you will need to implement a number of programs that access data in the designed database in order to support the functions of the system. The project consists of two phases:

1. Requirement Analysis, Conceptual Database Design, Technology Survey, Logical Database Design and Normalization.
2. Prototype Implementation

The first phase is to, based on the provided project description, analyze the requirements of **CanvasPath** in order to specify its system functionality and to identify data needed for the system functions as well as business rules (integrity constraints) to be imposed upon the needed data. In addition, it also requires you to come up with a conceptual database design by using the *entity-relationship* model to expressed the data and constraints identified. Furthermore, students shall perform a technology survey by researching on the current web/database application technologies,

including web frameworks, programming languages, development tools, and database management system, for comparison with those suggested by the CMPSC431W teaching team, i.e., *Flask Web Framework*, *Python*, *PyCharm IDE*, and *SQLite*. Additionally, logical database design and schema normalization is to be performed in the first phase. Each student will submit a well-documented report for the tasks in Phase 1. After this detailed design process, in Phase 2, students will populate the database and implement the system functions.

## OVERALL REQUIREMENTS

Before we jump into the description of each phase, let's talk about what you are expected to fulfill while working on the project.

### Document formatting

We will provide a document template (as .docx file) that you are more than welcome to directly follow. However, you are encouraged to create a template of your own, as long as it maintains a professional image for your cover page, and includes all the components from our provided template. You can also write your documents in LaTeX or other document typesetting languages or tools. Your report should be converted into one PDF file for submission **electronically**.

### Source Control

It is required that every student maintains a **GitHub** account and repository for the project. It is an excellent practice to maintain version control for projects of any size. Therefore, a GitHub classroom for CMPSC431W has been created for students to enroll. Note that the status shown on your repositories, e.g., how frequent the repository is committed may serve as an indicator for assessing your project effort and performance. In addition, you are required to maintain good documentation for your codes.

### Project Management

The success of this project heavily relies on your own personal effort and commitment. As denoted on the Syllabus and Course Schedule, the project is segmented into two phases. Thus, there will be no intermediate submissions. Students are expected to set goals and maintain individual milestones, documented as part of the Phase 1 report. For those who do not have prior background on web programming, it is highly recommended to start reading documentation and tutorials during Phase I.

Since there are numerous web application tools and frameworks, it is impossible for the teaching staff to provide guidance on every combination of tools. While students have a choice to use the recommended platform and tools (i.e., *Flask Web Framework*, *Python*, *PyCharm IDE*, and *SQLite*) which the teaching staff are more than happy to provide guidance, you may choose to adopt platforms, tools, programming languages and database management systems of your preference (but you are on your own in this case).

## PHASE I – DATABASE DESIGN AND TECH SURVEY

### Checklist

Tasks
Requirement Analysis
Conceptual Database Design
Technology Survey
Logical Database Design and Normalization

### Task 1: Requirement Analysis

As mentioned earlier, *CanvasPath* aims to replace the current course management system in Lion State University. Thus, it may help to explore the current Canvas (and other course-related) websites in PSU as a reference to better understand the expected functions and the requirements for *CanvasPath*. The following is a general description of the expected system functionalities in *CanvasPath*. In this task, you need to specify in detail the system requirements, including system functions, the types of data that must be stored, the business rules (integrity constraints) to be imposed on the data, and how it will be stored and accessed in support of the expected system functionality.

**Note:** For Task 1, you are expected to **elaborate** the system functionality in your own words and identify (also **write down**) the data needed to support the functionality as well as the integrity constraints to be imposed on the data.

In the following, we first describe the *CanvasPath* system from the users' perspective, then we describe some additional information in the system.

1. **CanvasPath Users:** All software and computer applications are to be used. Whether it be an embedded system which runs autonomously or a service/product which humans interact with, a good design needs to understand the purposes/goals of the systems and who the end users are. For this project, there are primarily two types of users, **Students** and **Faculty members**, as well as a special staff called **Administrator**, who takes charge of managing all students, faculty members, and courses information. At the beginning of each semester, the Administrator will create all courses on *CanvasPath*. He maintains all students and faculty members information such as the name, ID, age, gender, email address, home address, which includes street, city, state and zipcode, and initial login password. For students, the administrator also maintains their majors. For faculty members, the administrator also maintains their office addresses (different faculty members have different offices), titles and their departments as well. Then for each section of a course, he grants the faculty member who teaches the section the permission to manage the section. At the beginning of semester, the students enroll their courses but there is no opportunity for them to add or drop course after the semester begins. Note that the administrator only provides necessary initial information for course configuration in *CanvasPath* and he does not use it after the semester begins. Therefore,

*do not worry about the design of administrator.* The primary **CanvasPath** users whom you should be concerned about include only students and faculty members. We assume all users can log in with the email ID and the login password.

- a. **Students:** We need to be able to maintain information about students enrolled at Penn State. It is important that detailed information of students is collected and stored within **CanvasPath**.

For students, **CanvasPath** is a system to enroll into courses and to view their course information, assignments scores and courses grades. A student can log into **CanvasPath** with his email address and his own initial password at first. After logging in, he can modify his personal information (except for his ID) and reset login password. In addition, he should be able to browse/search the general course information (including instructor's name) for enrollment to courses. He can also find details regarding the courses that he takes (grades, instructor name and their details). A student cannot find specific information (e.g., assignments) of courses that he is not taking. For privacy concerns, students are only able to see their own grades. Furthermore, for a student taking a Capstone section of a course (Capstone section and project are to be introduced later), the system should also provide the student with contact information of their mentor/sponsor and the team members of his Capstone project.

Intuitively, for a student to be a valid user of the system, he must take at least one class, i.e. be enrolled to a course. In addition, a student can enroll to only one section of a course. Again, he can only view the specific information of courses he is enrolled.

- b. **Faculty Members:** As the same with students, faculty members can log in with their email and initial login password. After a faculty logs in, he can modify his personal information and reset his login password. Besides, he can see the list of the courses and sections he teaches. For a course section he teaches, he can see the names (along with IDs) of students enrolled in the section. If a section is not a Capstone section, the faculty can create entries for homework assignments and exams. He will submit the scores of assignments and exams for students after grading. At the end of semester, he will submit final grades for students based on their academic performance.

If a section is a Capstone section, the faculty member can create entries for homework assignments and some projects (which replace exams in regular sections). For each project, he collects the information of all teams (e.g., team name and information of team members). On **CanvasPath**, he assesses the performance of teams to assign grades for each team (i.e., he gives the same score to students in the same team). The faculty member can also see name and contact information of sponsors who mentor Capstone projects in the section. More information about Capstone section are to be explained later.

Intuitively, all course sections need to be taught by at least one faculty member. A faculty member may teach the same course in different semesters and he may teach multiple sections of the same course in one semester, which should be recorded by the **CanvasPath**.

However, only the courses (sections) being taught in the current semester can be managed by the faculty member.

**The above is stated from the user view of *CanvasPath*. The following parts describe the Departments, Courses and Sections which are to be managed in *CanvasPath*.**

2. **Departments:** Like all universities, departments in Lion State University are home of majored students and affiliated faculty members. Every faculty member belongs to only one department and all students must major in at least one department. For instance, a student may major in both of the Mathematics department and the Computer Science department (CMPSC), while Prof. Lee is a member of the CMPSC faculty.
3. **Courses:** Courses are the main area of focus which the *CanvasPath* project is targeting on. In Lion State, a course can only be offered by one department. In addition, courses have pre-requisites (one course can have multiple pre-requisites). In addition to pre-requisites, course information includes the course name, the unique number abbreviation, the department offering the course, and the course sections (you may use course information in LionPath of Penn State as a reference).
4. **Sections:** a section is the lower-level unit of a course. One course can have multiple sections. Sections are denoted by section numbers and each section is always taught by at least one faculty/professor. (Hint: Think aggregation!) Note that each section can be taught by a team of professors (i.e., faculty members). In such cases, professors teaching a course join some teams and each team takes in charge of a course section. Note that while the same course has different sections, each section has its own capacity limit, exams, homework assignments, i.e., Lion State University let professors (faculty members) to flexibly decide how many homework assignments and exams to give in their sections.
  - a. **Capstone section:** Some sections of a course are designated by a special type as CAPSTONE section. The only difference between a CAPSTONE section and a regular section is that in lieu of exams, students are to form teams to work on **Capstone projects** which are mentored by some sponsors. Mentors/sponsors are professors who are faculty members of any department in Lion State University. Every Capstone project has exactly one sponsor who may mentor multiple projects. The same project may be performed by different project teams independently, but they are all mentored by the same sponsor. Every Capstone project team, working only on one project, must have more than one student. As mentioned before, all students in the same project team receive the same grade, which is to be assigned by the instructor (i.e., the faculty member who teaches the section).

## Task 2: Conceptual Database Design

Based on the result of Task 1, you should present an entity-relationship diagram describing your conceptual database design. Also, your report should include a narrative description on all aspects of the diagram in detail. In addition to an overall ER-diagram of your conceptual design, parts of the ER diagram such as entities, relationships and integrity constraints are expected to be explained. Note that conceptual database design is application-oriented. Please do not assume the underlying database system (just yet) and thus do not map your design into relations.

### Task 3: Technology Survey

An aspect of working in the IT industry is to always keep your knowledge updated with the current market trends and technology innovations. Thus, for the project, it's important for you to have a good knowledge of the current web programming and database technology. A technology survey is meant for you to research a breadth of various web programming frameworks, programming languages, tools, and database management systems as well as the trends in those technological areas. Beside taking into consideration the web programming stack suggested by the CMPSC431W teaching team, i.e. Flask, Python, PyCharm IDE, SQLite, you have to come up with trending alternatives to make a comparison. It is expected for you to write persuasively about which tools you think as the best fit for the project. Please consider as many aspects as possible regarding the roles of those technologies in your project, and to *justify* your own recommendation by discussing the reasons, pros and cons of your choices. Also discuss the impact and relevance of those technology trends to the computer science fields and a broader segment of society or business/industry.

### Task 4: Logical database design and Normalization

Based on the Requirement Analysis and Conceptual Database (ER) design you developed in Task 1-2, you will finalize the relational schema for the ***CanvasPath*** database. You should produce a refined schema that reduces data redundancy to an acceptable level (i.e., the final schemas need to be at least in the 3<sup>rd</sup> Normal Form) while not unduly affecting performance. Your schemas should also support the enforcement of most, if not all, of the integrity constraints that you identify in this phase as well as those newly identified or added after relational schemas are generated from the ER diagram. **In this task, we assume two functional dependencies have been identified: a) zip codes in address determines state and cities; b) student email address which determines the login password.** You should present the specific details of how the process of schema generation and normalization is done, i.e., you are expected to apply both of translating the ER diagram to relations and then perform schema normalization.

## Grading

Different from exams, the project provides a way for students to show their commitment and effort in learning and practicing the knowledge learned from CMPSC431W. While the course project is work intensive, students who show their strong motivation and effort through the high quality of their reports will be rewarded. The project report, fulfilling the Writing requirement of the Penn State curriculum, is graded in terms of correctness, completeness, presentation, and clarity. Again, your effort will be reflected in your report and rewarded. Please be reminded that the project is work intensive -- start early and prioritize your time.

Extra credit of up to 10% will be rewarded for reports who include at least 1 new functionality expressed explicitly and clearly within the report. This is graded on novelty of the new functionality and clarity the student is able to present in various tasks of Phase 1.

## Submission

As mentioned above, a template will be provided as a .docx which serves as a mandatory style guideline. **The deliverable is a report in PDF containing the requirement analysis, conceptual database design, technology survey, and logical database design and normalization.** The document must have page numbers, section numbers and a table of contents. Figures should be used for illustration of your design. In addition, the document should also include a project plan (including schedule, deliverables, and milestones) as an appendix in the project report. It is expected that the report be at least 10 pages (**not including the cover page and table of contents**). Turnitin will be where you submit a .PDF version of your report. As stated in the syllabus, academic integrity is the most important and there will be no tolerance and exceptions towards violation. An incomplete but attempted submission is always far better than committing an academic integrity violation.

## PHASE II – SYSTEM PROTOTYPE

The EECS students have sent in their design proposals to Mr. Smith, who with the assistance of Dr. Lee and his teaching team, has determined the best conceptual and logical database designs. Now, Mr. Smith wants to validate the idea by prototyping a web application based on the selected logical database design and see how it work. To make it easier to compare different implementations, Mr. Smith has extracted a dataset (in the form of CSV files) from the legacy system, which will be used by the students to populate the database and test the prototyped system functionality. For evaluation, **the student implementations will be judged on the ability to execute designated tasks and the overall user experience from using the website.**

The schema for CanvasPath has been provided in a separate document. You are expected to follow this schema for Phase 2. However, you can make changes to the schema as long as you use the data from the CSV files we provide.

### Checklist

✓	Task
	Database Population
	Functionality Implementation
	Intermediate Review
	Reflections and Final Deliverables

## Project description

In Phase II, you will be provided with the following materials: i) a database schema design and ii) CSV files which contain the raw data. You are supposed to reorganize the raw data in accordance with the schema and populate the database. In addition to this, you will implement the web application for **CanvasPath** and demonstrate the functionality of **CanvasPath** as detailed in Task 2 below.

### Task 1: Database population

The first task is to create and populate the database. You will be provided with large data files where the columns contain values for various fields. However, please note that those are RAW data. In other words, do not simply treat those spreadsheets as ‘tables’ in the database since they may not comply to the integrity constraints. You are supposed to organize the raw values in the files based on the schema provided to create all of the necessary tables, define primary keys, foreign keys, and specify any integrity constraints as appropriate. In addition, define any views of the schema as you see appropriate. Again, please inspect all the columns and the corresponding



data types in the CSV files before you proceed to populate the database. *To support additional functionality you wish to add, you are encouraged to augment the schema and add your own data.* You can use Python libraries like Pandas to parse the CSV files and insert values to the table. You should provide (and maintain) one or more SQL scripts used in your data population, so that you can easily regenerate the database or restore the database states.

## Task 2: Functionality implementation

You should prototype (I.e., implement) the system functionality as specified below and be able to demonstrate the following operations on your final system prototype. The following is a description of the expected functionality.

1. **UserLogin:** This is for a user (a student, a faculty member) to log in. Note that the necessary users' log-in information is pre-stored inside **CanvasPath** system by the administrator. The system recognizes the user by his username and password. Please be sure that the user's **password is hashed when you store it** and is **not displayed while it is being entered**.
2. **CheckingInfo:** This is for a student to get information of the courses he is taking, which includes the course description, professor contact information (email and office) and course grades. If the section he is taking is a Capstone section, he will see the team number, mentor contact information and other team members information as well. Students should also be able to see their personal information, but **change password only**.
3. **Administration:** This is for an administrator (i.e., a university employee with the ID [admin@lionstate.edu](mailto:admin@lionstate.edu)) to manage the courses (add/remove), assign professors to courses, and enroll students to courses and sections. Ensure that no Professor is left unassigned and the students' enrollment to the course does not exceed the course limit.
4. **CreatingAssignments:** This is for a faculty member to create the assignments inside the section he is teaching. The assignments include homeworks and exams. The assignments will be seen by the students after the faculty member creates them.
5. **SubmittingScores:** This is for a faculty member to enter the scores of a certain assignment for all students after grading. There should be a grades page for each assignment which includes a table whose columns are the student's IDs and grades. The scores should be entered into the table. For uniformity, we assume all scores are out of 100.
6. **OrganizingTeams:** This is for a faculty member who is teaching a Capstone course to create a list of teams for the Capstone project. In his lecture, he collects the information of all teams (such as team members). The faculty member should maintain the team list and the members of each team on **CanvasPath**. When grading, students in the same team should be given same score.

**In addition to the functionality listed above, you may choose to implement one or more of the functionalities below for extra credits.**

Optional functionality to implement:

1. AssignLetterGrade: At the end of semester, CanvasPath will submit **final letter grades** for students based on their academic performance. Grades can be assigned with a 50-50 weight distribution for assignments and homeworks. You can use your own assumption for what final scores corresponds to which letter grade.
2. ReturnAverage: Students should be able to view the highest score, lowest score and average score in each assignment besides their own scores.

**Please note that while the correctness of functionality above is critical, the design of a user-friendly web interface is also very important to the success of the prototype. As we expect this system to impress the members of the Lion State Advisory Board and eventually be used by all the students and faculty members of Lion State University, please pay attention and effort to the web design of the prototype.**

### Task 3: Midway Review

We will have a midway project review to serve as the progress checkpoint before the final deadline. It will be held during *the first week of April* and you are expected to complete and showcase to the teaching team *at least* the following tasks:

- Database population
- UserLogin

You will be given 5 minutes to demonstrate the above. We will be post the review schedule and signup sheet for your respective project demos at a later time. Please note that we expect you to use GitHub for project management and we will be looking into your GitHub commits as part of the review.

### Task 4: Reflection

At the end of project, it's beneficial for you to flash back what you expected at the beginning of the semester regarding the project and reflect on whether you meet the goals you set back then. Make a comparison between what was expected, and what has been achieved. Discuss the lessons learned – for example, why some parts of the required functionality were not completed (if any), how you added some new features (and why, if any), how you would proceed with the project the same or differently in the way you do in the course, and your thoughts about the project and suggestions to Mr. Smith, etc. Please plan and include the reflections as part of your video presentation (see more details of the video presentation in the Deliverables section below).

### Final Deliverables

The deliverables in Phase 2 include **a video presentation and a demonstration of the CanvasPath system.** In your video presentation, please discuss important design and implementation details you considered and make a reflection. Your video presentation should be submitted to Box via Canvas. For help with making the video presentation, please have a look at: <https://mediacommons.psu.edu/support/tutorials/>

In the project demonstration, you are expected to show your prototype implementation of the *CanvasPath* system to the teaching team. We will use the website and check some/all of the compulsory tasks with data from the CSV files provided to you. You may also show the optional functionality for extra credit. Please be well-versed with your code as we may ask you to explain the web server code and the queries you wrote.

**As stated in the syllabus, academic integrity is the most important -- there will be no tolerance and exceptions towards violation!** An incomplete but attempted submission is always far better than committing an academic integrity violation.

Your efforts will be rewarded with extra credits!