University of Washington

iSchool Info 340

# Info 340 - Final Project

In this Final, you will plan, document, and implement a database used by an application being created by another development team.

The purposed application consists of a simple internal windows application that handles the act of scheduling appointments between patients and doctors.

This application allows a use to:

* Search for existing patients
* Add a new patient
* Search for doctor
* Select a clinic
* Select a date and time

## Application Flow

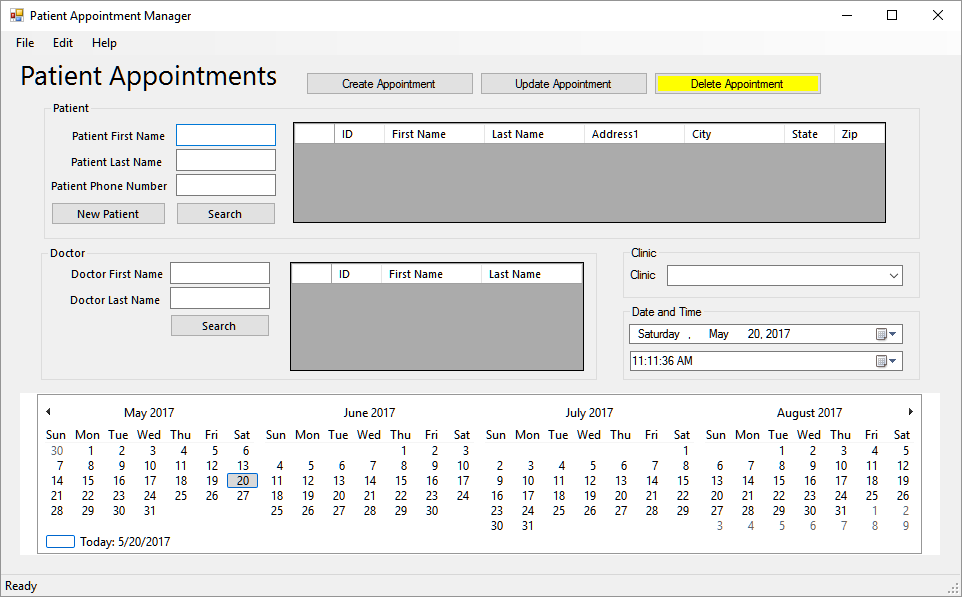
1. User looks up an existing patient (If patient cannot be found a new record can be created).

2. User selects a doctor (or clinic)

3. User selects a clinic (or doctor)

4. User selects a date and time for the appointment

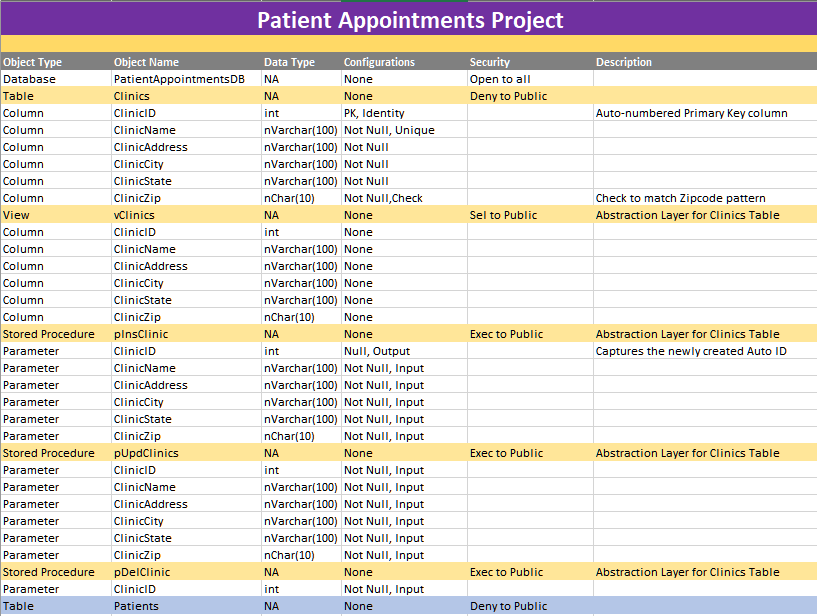
5. User Creates the appointment (or updates appointment or delete appointment)



## Database Design

You need to design a database based on what you have learned about the application. This will be a prototype, so expect there to be issue that you will have to resolve or note. When you find issues, or have question for the Application team, you need to note these for discussion once your prototype design is complete.

You will use an Excel Spreadsheet to document your design. I have provided a starting file for you to use and have filled in the metadata for the Clinics table and its abstraction objects. You need to complete the spreadsheet with metadata for the other tables and their abstraction objects.



## ERD

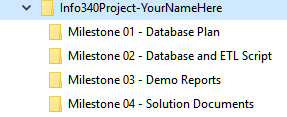
After you have filled in the Excel spreadsheet, create a database diagram (ERD). Take a screenshot of the ERD and paste it into a Word document called, "PatientAppointmentsDB.docx."

## Task Checklist

Your grade will be based on how well you do on the following tasks (in addition to the general grading rubric!)

1. Create a development documentation with Excel (**80** pts)
2. Create an ERD as an image pasted into Word (**20** pts)
3. Create a Database Script (Must run from Start to Finish without errors!) (**200** pts)
   1. Use the STARTER\_FinalDB.sql file for your code.
   2. Rename the file, by replacing STARTER with your name.
4. Create a separate SQL Script to import data using a series of Insert statements. Include 10 clinics, 100 Doctors, 1000 patients, 1000 appointments (**20** pts).
5. Create a Reporting View (10 pts)
   1. You should already have one reporting view in your database, called vAppointmentsByPatientsDoctorsAndClinics. That view shows data from all the tables and can be used for your reports.
6. Create an Excel Reports (**20** pts)
   1. You will need to create an Excel report using data from a view. The report should include one “table like” display of data and one chart. It must also include a Title and header with your name and the date.
7. Create a Power BI Reports (**20** pts)
   1. You will need to create a Power BI report using data from a view. The report should include one “table like” display of data and one chart. It must also include a Title and header with your name and the date.
8. Create a Lessons Learned document (**20** pts)
   1. This is a relatively informal overview of the different things you have learned during the quarter. It does not have to be an exhaustive enumeration of these items but should include at least two or more different aspects of creating and working with a relational database.

***NOTE****: Take time to place all your files into the folders I provide.*



# Grading

Student work will be evaluated on a point system using the following general guidelines:

|  |  |  |  |
| --- | --- | --- | --- |
| ***GRADING RUBRIC: GENERAL ASSIGNMENTS*** | | | |
| **CRITERIA** | **EXCELLENT (90% or more of full points)** | **PASSING (60% or more of full points)** | **NOT PASSING (less than 60% of full points)** |
| **DEMONSTRATES COMPREHENSION OF CONCEPTS** | Your submission shows firm understanding of assignment concepts. You’ve internalized the principles, and adapted them to your own relevant work. | Your submission shows a general understanding of assignment concepts. You might have a few areas to improve upon. | Your submission shows little effort to understand the assignment concepts. |
| **CORRECT ORGANIZATION** | Correct organization of files, code, and structures throughout submission, with maybe a couple of small errors. | Mostly correct organization of files, code, and structures throughout submission. | Mostly incorrect organization of files, code, and structures throughout submission, with little effort demonstrated. |
| **CORRECT OPERATIONS AND FUNCTIONALITY** | Correct operations and functionality throughout submission, with maybe a couple of small errors. | Mostly correct operations and functionality throughout submission. | Mostly incorrect operations and functionality throughout submission, with little effort demonstrated. |
| **PROFESSIONAL PRESENTATION AND COMPLETENESS** | College-level or better presentation and formatting of materials (document have titles, screenshots have captions, etc.) Spelling and grammar has few errors. | Mostly coherent presentation and formatting of materials, with the most important pieces of information present. Some spelling or grammar errors. | Missing or incoherent presentation. Documents missing student’s name, date or titles; poor spelling or grammar. severely incomplete work. |

# Submit your work

After you have completed all of the tasks and placed all of the files in their folders, zip the parent folder (“Info340Project-YourNameHere”) into a single file and upload it to Canvas.

**Tip:** Don’t forget to change the “YourNameHere” to your own name!

# You are done with Final!