Angeline Ortiz de los Santos

5/18/2024

ECPI University

**2.6 Course Project Application Design**

* Final/updated project proposal

## Application Description - a paragraph (3-4 sentences) explaining what your application is and what it is supposed to do.

My application will be a system to help Juvenile Probation Officers facilitate decision making for their community base programs. This application will allow Juvenile Probation Officers to determine what program is most suitable for a youth in the juvenal justice system. Also, insights into the success of each program will allow officers to determine what programs are worth funding and what is declining in the success rate.

## Purpose - explain the purpose of your application - why is this something that should be created, and what value does it provide?

This application will bring incredible value to the juvenile justice system since probation officers depend on the success rate of these programs for the juvenile to stay in the community instead of being incarcerated with no treatment to prevent juveniles to offend in the future.

The data from my application will be beneficial to probation officers because it will help determine if that community-based program is appropriate for the child’s needs and treatment based on the type of crime. Also, it helps determine the availability of each program. This information will be helpful to understand If the programs are having a high success rate and what they do to achieve this.

## Output - explain what type of output/information your application will output to the terminal window

My application will be able to output the price per child per program and curriculum. Also, youth information for example attendance, and grade. How many children were able to complete the program vs how many did not. Also, how many infractions the child committed during the program.

## Data Storage - describe, generally, what data your application will store

My data application will store two main portions, students and program. For the student we will have information such as demographics and program success. For the program, we will be able to store program information, price, curriculum, qualifications, and availability.

* Class definitions
  + Class Name: Program, Cost, Juvenile, and Success Rate
  + Properties:
    - * Program:
        + +ProgramID: int
        + +name: string
        + +Location: String
        + +Shelter: bool
        + +SecureFacility: bool
      * Cost:
        + +ProgramID: int
        + +CostPerMonth: int
      * Juvinile:
        + +juvenileID: int
        + + ProgramID: int
        + +name: string
        + +Age: int
        + +CrimeType: String
      * Success Rate:
        + + juvinileID: int
        + + ProgramID: int
        + +grade: string
        + +attendance: int
        + +graduate: bool
  + Methods: +GetProgramInfo();
  + Special considerations (output format; how output will be provided - ToString, separate method, etc.; class hierarchy - inheritance, interface, polymorphism, abstraction, composition; access specifiers)
    - * class hierarchy - inheritance,
  + UML diagrams are not required but are highly encouraged. If you use UML class diagrams, you must still include textual descriptions of the classes, but they can be much shorter because the diagram will show most of the information.

**UML Diagram**

(abstract)

Crime

+CrimeType(): String

TreatmentProgram

+ProgramID: int

+name: string

+Location: String

+Shelter: bool

+SecureFacility: bool

+CrimeType(): String

+Cost(): double

+ToString()

---------------------------

+GetProgramInfo();

Juvenile

+juvenileID: int

+ ProgramID: int

+name: string

+Age: int

+CrimeType(): String

+ToString()

SuccessRate

+ juvinileID: int

+ ProgramID: int

+grade: string

+attendance: int

+graduate: bool

+CrimeType(): String

+ToString()

* Data storage
  + Table name(s):
    - * Program, Cost, Juvenile, and Success Rate
  + Fields the table will contain, including data type:
    - * Program:
        + +ProgramID: int
        + +name: string
        + +Location: String
        + +Shelter: bool
        + +SecureFacility: bool
      * Cost:
        + +ProgramID: int
        + +CostPerMonth: int
      * Juvinile:
        + +juvenileID: int
        + + ProgramID: int
        + +name: string
        + +Age: int
        + +CrimeType: String
      * Success Rate:
        + + juvinileID: int
        + + ProgramID: int
        + +grade: string
        + +attendance: int
        + +graduate: bool
  + Table relationships, if any:
    - * Program will have a one-to-many relationship with Cost and Juvenile. Then, Juvenile will have a one-to one relationship with Success Rate.