

CS3520 Programming in C++

Final Project

Objective:

Design, develop and implement a working software system in C++. All code must be maintained on [GitHub](#). Please be sure to review:

- Problem Statement
- FAQs
- Final Project Rubric

A Word of Caution!

*You have tremendous amount of freedom in designing this application. You are welcome to add any additional functionality to your system. Just imagine that you are an actual developer of the application. My suggestion will be to design a version with a basic set of required features {listed in this document **and rubric**} and avoid over-complicating the design*

Problem Statement – Court Reservation System

Congrats! City of Newton has just hired you to create a system that will be used to manage reservations for their new Pickleball facility. This facility has 3 courts that may be reserved by members. Since courts are shared by coaches (for training sessions), players (for individual reservations or group game sessions), we need to design a system that can help maintain the schedule. This semi-visual system will help us see if courts are free for a reservation or already being used by someone. For this course project, a textual Reservation System would suffice but you should feel free to use any GUI library (if you want).

Here is a sample view from a similar system used for a local Badminton club. Other examples include clublocker.com, courtreserve.com, etc.

	Badminton Court #1	Badminton Court #2	Badminton Court #3
6 AM	6:00 AM(30m) Past	6:00 AM(30m) Past	6:00 AM(30m) Past
6:30	6:30 AM(30m) Past	6:30 AM(30m) Past	6:30 AM(30m) Past
7 AM	7:00 AM(30m) Past	Tao Han <Open>	Kim Cha Lim <Open>
7:30	7:30 AM(30m) Past	<Open> Ling Bai	<Open> Tsung Lin
8 AM	Past	<Open> Shuo Zhuang	<Open> Peng Lu
8:30	Clinic - Wee		
9 AM	5 open spots		
9:30			
10 AM	B/C Drop in 46 open spots	B/C Drop in 46 open spots	B/C Drop in 46 open spots
10:30			
11 AM			
11:30			
12 PM	Sasikala Jayalekshmi <Open>	12:00 PM(30m) Past	12:00 PM(30m) Past

Here are some important details regarding our system:

- **Courts:** The facility has 3 available courts, and the facility opens from 6am to midnight. Coaches are allowed to hold training sessions or lessons only during 9am-Noon and 3-6pm on weekdays. Club officers may schedule open-play times (for group sessions) on one or more courts for a specific category of members on any day from 6-9pm. Coaches are expected to schedule their coaching sessions atleast 48hrs in advance otherwise, courts become available for members to reserve. Same applies to open-play sessions. If the courts are not being used for open-play or by coaches, then individual members may reserve the court for themselves.
- **Reservations:** Each reservation slot is 30 minutes long. A player may not reserve more than 30 minutes per day and may not reserve more than 1 hour of court time per week. Each time slot may have a maximum of two players (where second player can join the reservation of the first player or a reservation may be held by a single member). If you join a reservation as second player, it will be counted as 1 reservation for the day (and the week). Reservations may be deleted only by the players on that reservation (or by officers upon request). Coaches don't have reservation limits as other members.
- **Users:** There are three kinds of users - club members, club officers and club coaches. Members will be able to use the system based on their roles (or types). Club officers are also considered club members (while coaches are not).
 - **Club Coaches:** Coaches can reserve any of the courts for individual lessons or group training sessions during the designated coaching hours only. Coaches have priority in reserving courts during coaching hours if the reservation is made more than 48hrs earlier. Otherwise, they can only reserve courts if courts are still available (i.e., were not reserved by someone else)
 - **Club Members:** Club members are players who play pickleball during open-hours or during the times they have reservations. Club uses a 3-scale rating system (ABC) for players' skills, where best players are rated 'A', average players are rated 'B' and so on.
 - **Club Officers:** Officers are members with the additional responsibility of serving as officers. Officers can reserve courts for open-play hours during designated time slots. Officers also can modify the reservations from players or coaches upon their requests (i.e., make reservation for them or modify or delete the reservations).
- **Reservation System:** Reservation system offers different features to the members depending on their roles (or types). It must strictly enforce rules (as to the number / time of reservations). Reservation can be made up to 7 days in advance. Any user should be able to see the daily schedule for all courts (see image above) where each 30-minute slot shows the name(s) of people on that reservation. Open-play sessions are utilized by players with the same specific skill level (or rating) but they are not counted towards member's reservation limits. If a court has not been reserved (by coaches for training or officers for open-play) during designated times atleast 48 hrs in advance, then its reservation can be made by anyone on First Come First Serve basis.

System Features:

The software you are designing is intended to function as if it was actually going to be used by the City of Newton, thus the user experience you create must be realistic.

Here are some required functionalities:

- Each user should have a unique username (or id) and a specific membership type in the system. At the start, user is presented with a set of menu style options for different features such as view today's schedule (all users), send a message to an officer to request cancellation or modification (members and coaches), reserve a court (all users, where each user may see different available time-slots depending on their role), cancel my reservation (all users), cancel others' reservations (club officers only), and other features as appropriate
- The system should be user-friendly, so there should never be any confusion about what a certain action will do and what input is expected from the user. This means you will probably have to print instructions to the user anytime they are expected to provide any user input. Provide sufficient guidelines and help to use the system (For example: if your system is expecting any input in a specific format, be sure to specify that in the instructions as well as the readme file for your project submission).
- Your system must have **state persistence** by saving the necessary data to files. State persistence allows your system to outlive the process that created it. For example, if you add a member and/or reservation to the system when you initially run it and then close the execution before running the system again, the system must be able to remember the previous reservations and its state. This will allow you to use the previously entered information without having to create all users, reservations, etc. again and again. You can complete the entire project by storing data in files or any other persistent storage medium

Note: The design of a Graphical User Interface (GUI) is not needed and is optional. You should develop a command-line based system.

Guidelines:

- You may work in groups for this project (up to 3 members maximum). You **MUST** Self-join a group from Canvas -> People -> Groups4FinalProject after coordinating with your group partner or join an empty group if you are working alone. DO NOT CREATE A NEW STUDENT GROUP YOURSELF, ONLY JOIN ONE OF THE AVAILABLE GROUPS.
- If you have an unwanted person in your Canvas group, please contact the instructor to have them removed!
- See the deliverable section below for more information

Design & Development:

The final project includes two major components: Project Design and Implementation. You **MUST** design the software BEFORE starting on any code writing.

Design: The design component will consist of the software design for the project and will include the list of all classes (member variables and functions) to be used, their relationships & collaboration (CRC cards) and UML diagrams (Class diagrams and sequence diagrams), as well as data structures/databases/files. The initial design must precede any implementation. You will start by analyzing the requirements of the project and by identifying the classes required along with attributes and functionalities.

Make sure that your design follows good object-oriented design principles such as **SOLID, DRY, KISS, YAGNI**, and so on. A doc/docx document listing the design along with above listed diagrams must be submitted by the deadline on Canvas. Please include the following **in your report**:

- Class Diagrams with all classes (with member attributes and member functions), and CRC Cards

- What are class hierarchies and relationships? (UML Diagrams)
- Sequence diagrams (for most important features)
- List of all other data structures or files to be used

After the first version of the initial design is completed, it would be beneficial for you to analyze your design thoroughly and try answering some of the following questions:

- How many classes will you have and how will they interact?
- How will you store user information?
- How will you use hierarchy for different user types?
- How will you represent the project/task selection options and the menu?
- How will you deal with weekly limits on reservations?
- What information is needed to make your system state-persistent?

Initial version of the design will be due on Canvas by **Thursday 6/22**. *You will have an opportunity to discuss your initial design with TAs during office hours.* You can update the initial design to incorporate any feedback before starting your implementation.

Now imagine yourself as the user of the system (as club member and/or coach) and observe the behavior of the system (and see how it changes). Reflect on the efficiency of your choices. Review your design and make the necessary changes to remove any unnecessary attributes and functions. You may have to add some more functionality that you might have missed in your initial design. Your **Final Report** should be a doc/docx/pdf document listing the final design.

Implementation: Once you complete the design, setup a *private* github repository (on Khoury github) and start implementing your project. Be sure to add the appropriate user-friendly menu options that would allow the user to select various actions easily. Ensure that the menu options only work when correct input is provided by the user. Test each option individually to ensure that all available functionality is properly implemented. **All sourcecode must be shared amongst team members via github.** Don't forget to add us (TAs and instructor) as collaborators.

Testing: Test each operation and each component of the system as a member, coach and office. Please ensure that the complete system is properly tested before submission. Use the rubric as a check-list (if needed). Here is a list of tests that should be part of your overall testing:

- Your software can display the daily court schedule for all users. The most up-to-date schedule should be shown to all users.
- You should be able to reserve the court, delete reservation, join reservations, remove yourself from reservation, request officer to delete or modify reservation and so on.
- Your software should ensure that you can exit the application at any time and can re-run it without losing any information (reservations, members and their status).

The final implementation along with final port are due on Canvas by **Monday 6/26**. Be sure to review the **checklist** given below:

Deliverables: The following items must be submitted:

- **All source and header files** related to the system implementation (on Khoury github).
- A **README** file with any information regarding compilation and testing that we may need to know, in order to compile and run your software successfully. Include **any other files** needed to compile or test or any default username and passwords.
- **Source code must be properly organized, readable, and must use proper best coding practices.**

- **The report** with the final design (UML diagrams showing class relationships & hierarchies, and sequence diagrams), CRC Cards and brief details of testing activities. Feel free to show additional testing you performed in the report
- {Optional} Video Demo (12 minutes long max) showing all features of your working system. Voice over is not mandatory

Be sure to use github properly i.e., to store all updates of your work throughout the project. Make frequent commits to github to show your progress and avoid using github for a single, final commit only. After the work is completed, commit the final version of the code to github, then download it again (on a new computer) to ensure that it compiles and runs as expected.

Additional Requirements:

- Properly implement all constructors and destructors (Rule of 5). Use RAII design principle.
- Use STL for any data structures, and algorithms that you might need.
- Prefer using smart pointers over raw pointers.
- If using raw pointers, use dynamic memory allocation as much as possible and make sure that your software does not have any memory leaks (check with valgrind to confirm and provide screen grab)

Submission Checklist:

You can use the following checklist to ensure that you have submitted everything required for the project:

- {Before submitting your commit link}, did you download the final version of your code from Khoury github to verify that it works?
- Did you submit the instructions required to test your code in a readme file?
- Does your submission include your report (with UML diagrams), and the names of your partners?
- Did you add the instructor and TAs as collaborators to your private github repository?
- Did you submit *optional* video demo?

FAQs:

Here are some common questions

- **Q:** Who adds the members or coaches to the system? How many members should there be?
A: At startup, you can add few users as needed. At the minimum, you should setup your system with 5+ users, 2+ officers and 2+ coaches.
- **Q:** When do we create users? How do you switch users? Do they need to login?
A: There are two options that are typically suggested: one is to create a few users (members, club officers and coaches) at the start and the other is to create them on the fly. Use whatever works for you. In either case, their information should be preserved. It would be advisable to implement a login screen/menu where users can provide their username and password. It would make sense to add those as options to main Menu.
- **Q:** How many days should a user be able to view the schedule for?
A: User should be able to see the schedule for the entire week. You may also provide an option to show the daily schedule or schedule for 3 days. You should populate data for atleast 2 weeks. Be sure to add realistic test data instead of using dummy names.
- **Q:** Do we display real time or days? When does the week start?
A: Yes, you should display real weeks (may be even real day/time). You can start the week from Monday or any other day.
- **Q:** What happens when a member requests an officer to delete a reservation?
A: To keep things simple, an officer can delete the reservation upon request. This request may be coming through a message from members or coaches. Similarly, officers can make a reservation requested by coaches or other members.
- **Q:** What are open-play hours and how does it impact reservations?
A: Open play hours are available for players to play with other players of similar skill set. Players gather on destinated days and start playing. Our system does not track who is playing and for how long and that's why it is not counted towards daily or weekly reservation limits.
- **Q:** What happens if a coach cannot request a court atleast 2 days earlier?
A: If the coach has not reserved the court in advanced, it becomes available for other players to reserve for individual play. Coach can still reserve court within 48 hrs only if the court remains unreserved. Same logic applies to open-play hours.
- **Q:** Can coaches reserve courts outside coaching hours? Can open-play sessions be outside specific hours?
A: No and No. Coaches are not members and they don't get to reserve courts like regular players. Coaches can only coach during training hours. Similarly, officers can only reserve courts for open play during specific hours.

Initial Design of Final Project: Reservation System - (25 pts)

General – 25 pts

Criteria	Pts	Description
Initial Design	10	-10: UML diagrams [Class Diagrams showing relationships and Sequence diagrams], CRC Cards, and brief discussion -5: Some diagrams are missing, or they don't include important information
Quality of Design	15	-15: Quality of Initial design (following Design Principles) -5: Minor design flaws

Final Project Rubric (150 pts) – Reservation System

General – 40 pts

Criteria	Pts	Description
README file	0	-2: If necessary, info to run the program is missing
Final Report	10	-5: UML diagrams [Class Diagrams showing relationships and Sequence diagrams], CRC Cards missing -5 Discussion of Design not included
Quality of Design	20	-10: Quality of Final design (following Design Principles) -10: UML diagrams [Class Diagrams showing relationships and Sequence diagrams], CRC Cards, and brief discussion
Video Demo	0	Should showcase all features of working system
Coding Style	6	-1 for each deviation (missing header block, missing indentation, improper class, function, and variable names, adequate comments, etc.)
File Organization	4	-4 File organization
Makefile	-	Automatic deduction of 50% (minimum) if the code doesn't compile. TA will try to give partial credit by visually inspecting the code.

Project specific – 110 pts

Criteria	Pts	Description
members types and member info (unique username and details).	10	-8: If different types of users are not supported (i.e., Coach / Players) i.e., Hierarchy of Users -2: The program does not prevent duplicate usernames and does not provide appropriate error messages
view daily schedule and reservations	17	-7: Court availability is not maintained properly. -5: The program does not neatly display updated schedule for all courts, showing the name(s) of people on each reservation. -5: If users can't check and track their reservations for the day or week.
Members can reserve a court	23	-3: Club coaches should not be able to reserve any court for individual lessons or group training sessions other than designated coaching hours. -3: Club members should not be able to reserve courts other than their designated reservation times.

		<ul style="list-style-type: none"> -3: Club officers are not able to reserve courts for open play during designated time slots. -2: Club officers are not able to reserve courts for themselves. -2: 3-scale rating system (ABC) for players' skills is not implemented properly. -3: Officers are not able to modify reservations from players or coaches upon their requests. -2: Member can't join another reservation. -3: Deletion of reservation is not proper by all members. -2: members can make reservation during restricted hours (open-hours or coaches training slots) if they were not reserved 48 hrs in advanced
System enforces reservation rules	16	<ul style="list-style-type: none"> -2: Reservation slot length is overlapping. -2: Maximum reservation time per day is not maintained. -2: Maximum reservation time per week is not maintained. -2: Maximum number of players per time slot are not maintained. -2: Counting second player reservation. -2: Open play counted towards members reservation limit. -1: "Reservations can be made up to 7 days in advance for officers and coaches" is not enforced. -1: "Reservations can be made up to 2 days in advance for officers and coaches if the slot is not booked by any user" is not enforced. -2: "Reservations can be made up to 2 days in advance for other users (if officers and coaches have not booked the particular slot)" is not enforced.
cancel others' reservations	5	<ul style="list-style-type: none"> -2: Members should request cancellation of their reservation by sending a request to the officers. -3: Officers are not able to cancel request for each user type (coaches/individual players)
Overall Quality of Program	11	<ul style="list-style-type: none"> -2: If program lacks logical menu options -2: Ability to switch to different user -2: If user is not given proper guidance or useful prompts for inputs (Usability) -5: If user experience is not delightful or fun.
Data persistence	15	<ul style="list-style-type: none"> -5: for each instance of lack of data persistence. (All info for players (skills, weekly/Daily booking should be retained). -10: User should be able to successfully save the state and load it back without losing any data.
Memory Allocation & Using smart C++.	13	<ul style="list-style-type: none"> -3 Implementation of RAII is not proper. -3. Rule of 5 is not implemented. -3: STL Not used in the code. -3 All Memory not released (if raw pointers are used), try to use smart pointers. -1 If screen grab of <i>valgrind</i> not provided
Compile and Run		-50% if program doesn't compile or crashes. We will test it on Khoury server only.

Change Log:

- March 26th (7pm): Initial Version
- April 17th (11am): Added FAQ for coaches' reservations.