Arshdeep Kaur (A 01)

200465607

Software Architecture Document

**Architectural Representation**

* Architectural Factors
* Performance/ Accuracy
* Tee Time Membership levels
* Player Scores

Performance/Accuracy for TeeTime Membership levels should be high because of the high business value. Also, Performance/ Accuracy for PlayerScores should be high because Players are the potential clients, and so calculating scores accuately is important.

**Architectural Factors**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Category/ Subcategory** | **Factor** | **Measures** | **Variability** | **Impact of factors** | **Priority of Success** | **Risk** |
| Performance/ Accuracy | Tee time membership levels | Gold- Anytime  Silver-  Mon-Fri  (Before 3 p.m., after 5:30 p.m.)  Weekend/Holiday  (after 11 a.m.)  Bronze-  Mon-Fri  (Before 3 p.m., after 6 p.m.)  Weekend/Holiday  (after 1 p.m.) | * Current flexibility (levels- gold, silver, bronze) * Future evolution/ expand | On Stakeholders, Players, Clerk, Pro-shop staff, Membership Committee | High | Critical |
| Performance/ Accuracy | PlayerScores | Handicap | Change in Handicap over time | Players | Medium | Significant |

**Architectural decisions**

Technical Memo

Issue: Scheduling tee time for membership levels and Player Scores

Solution Summary: Using Identity Framework for Membership levels and separate table for Player Scores

Factors: Tee Time Membership Levels and Player Scores

Solutions: Identity Framework is used for Membership levels and they can login and access into their accounts or book their tee times. In future, if there will be any additions in the Membership levels, levels can be added easily by adding any number of users and giving access to those through login user id and password. Players can add their score each time they play which will be saved in Player Scores table. This table will also include Course field which allows player to add scores even if they play a different golf course.

There will not be any other major modifications that need to be made in the system and which will prevent breaking in other parts of the system. System will perform as usual.

Motivation: Using Identity framework will allow expanding membership level to be much simpler and easier. Having table for Player Scores will allow flexibility to add Player Scores with the entry of their Course and keep record each time player plays

Unresolved Issues: N/A

Alternatives considered: The other alternative may be that we could have created separate table for Membership levels and Types and if there would be needed to add more levels, we just had to insert new row into the table.

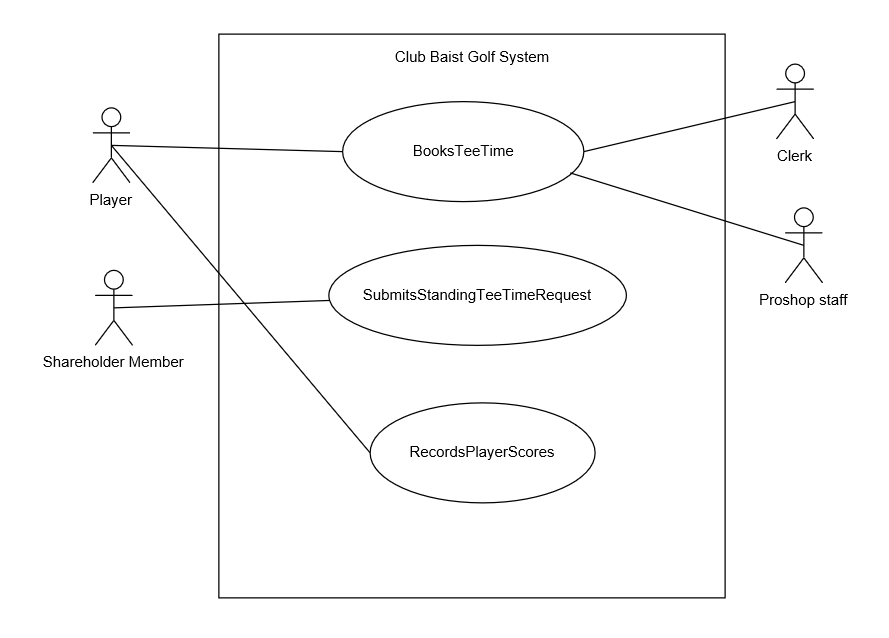
* Use Case View
* Use Case View

**Architectural Views**

* Use Case View

*Architecturally significant use cases*

* Books Tee Time
* Submit StandingTeeTimeRequest
* RecordsPlayerScores



Discussion and Motivation

BooksTeeTime

The purpose of this use-case is to allow players, clerks and pro-shop staff to book tee times.

Different levels of player can book at different times.

Can be done:

By Players- Gold, Silver and Bronze through online website 1 week in advance,

By Clerk on phone, 1 week in advance,

By Pro-shop staff in person or by phone, the same day golf is played.

SubmitStandingTeeTimeRequest

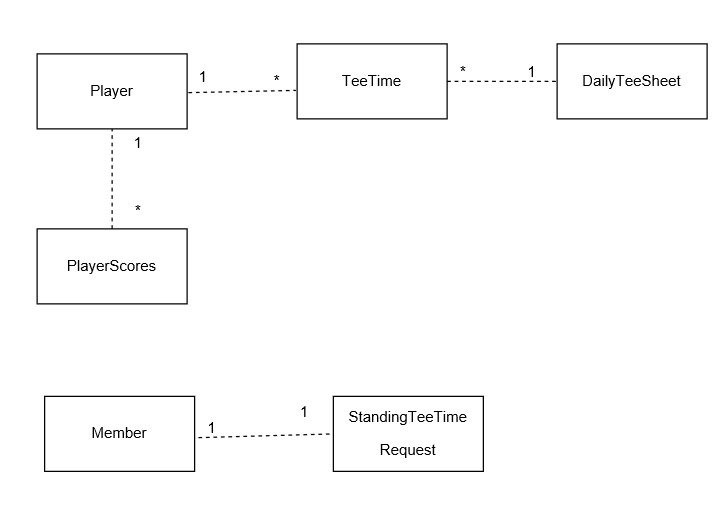
The purpose of this use case is to allow shareholder members to make standing tee time requests.

RecordsPlayerScores

The purpose of this use case is to allow Players to record their player scores each time they play and the also the value at end of eaach month. Also, they can add their scores even if they play different course.

* Data View

Architecturally Significant data elements



Discussion and Motivation

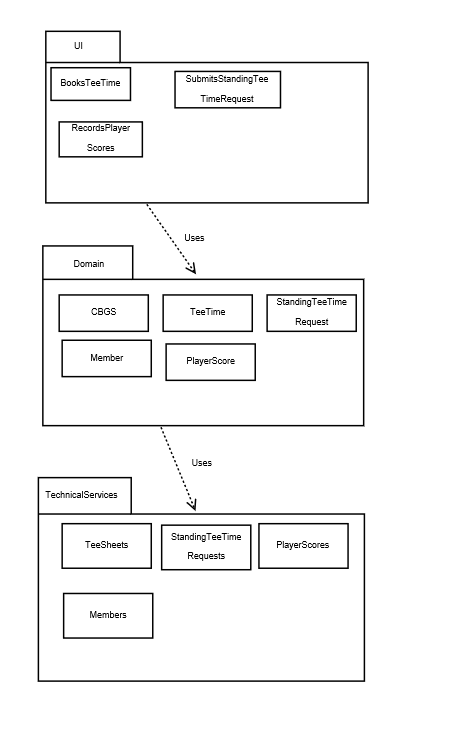
Player can book multiple tee time, that means any Player can appear multiple time in Tee Time table but that Player has to have in Player table first. Daily Tee Sheet has Date which acts as a primary key in Daily Tee Sheet and acts as a Foreign Key in Tee Time table, means date in Daily Tee Sheet table can appear multiple times in TeeTime table but has to have in Daily TeeSheet Table. Also, player is associated with Player Scores. Player can have multiple entries in PlayerScores table, where each time Player plays, his scores are recorded.

Any Shareholder member can submit one standing tee time request, this why the relationship between Member and StandingTeeTimeRequest is one to one.

* Logical View

Layers, Architecturally Significant Design Classes

*Layer Diagram*

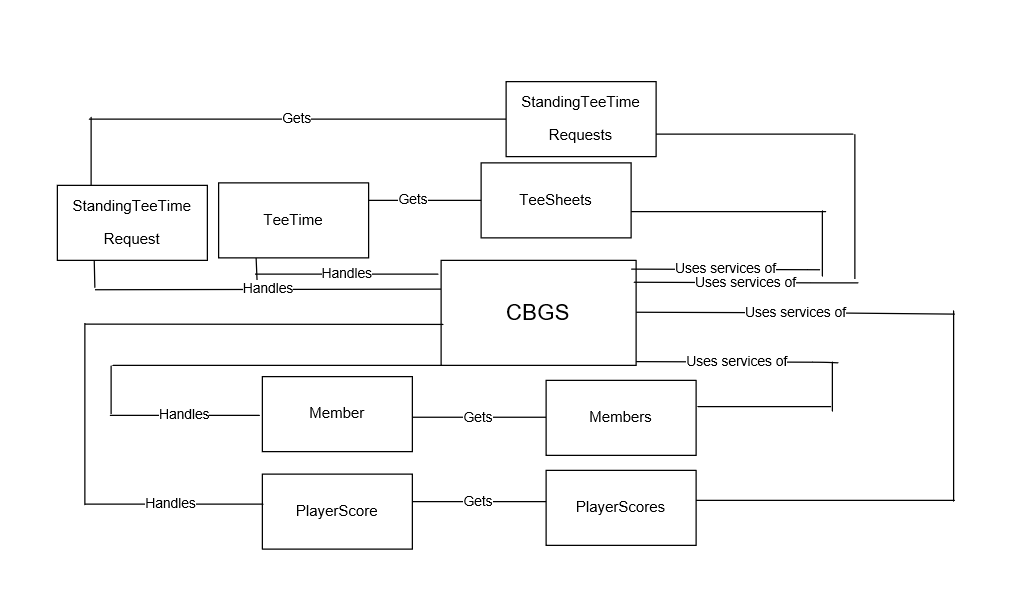


Discussion and Motivation

There are three layers i.e. UI Layer, Domain Layer and Technical Services Layer. UI Layer which contains all the pages that appears as the User Interface comes under UI Layer. These are basically razor pages. So, here we have Razor pages or forms that appears to the users and users can BookTeeTime, SubmitStandingTeeTimeRequestRecordsPlayerScores. This layer interacts with the Domain layer which contains the classes such as TeeTime, SubmitStandingTeeTimeRequest, Member, PlayerScore and CBGS. UI Layer uses classes in Domain Layer and passes request to the Domain layer which then uses classes in Technical Services Layer. Technical Services layer contains classes which are used to call the procedures that are required.

*Design Class Diagram*

Architecturally significant classes

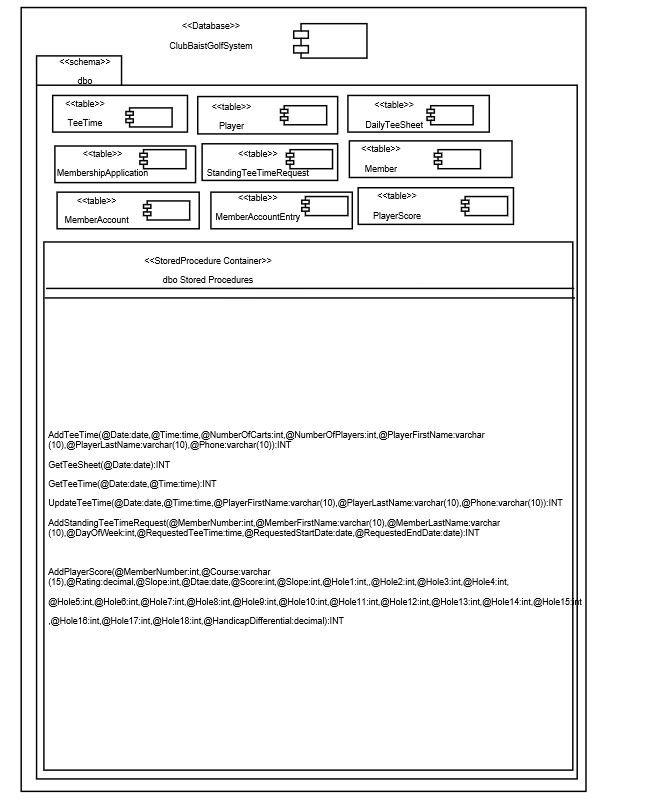


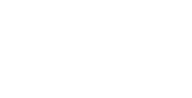
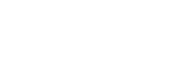
Discussion and Motivation

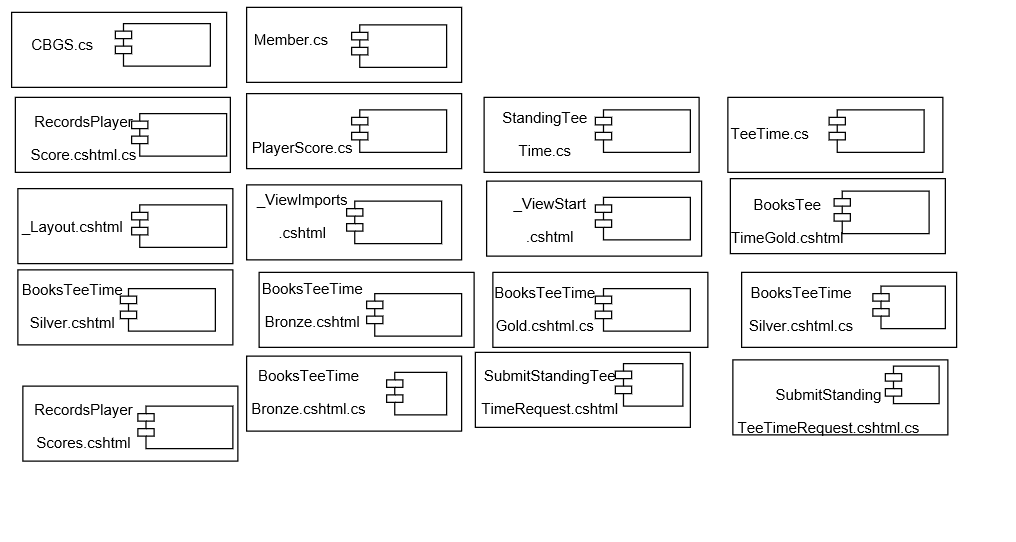
CBGS Class acts as the RequestDirector which uses services of TeeSheets gets TeeTime and Handles TeeTime to the CBGS or uses services of StandingTeeTimeRequests to get StandingTeeTimeRequest and handles it to CBGS or uses services of MembershipApplications or gets MembershipApplication and hanles it to CBGS or uses services of Members, gets Member and handles it to CBGS or uses services of MemberAccounts or gets MemberAccounts and handles MemberAccounts and MemberAccountEntries to CBGS or uses services of PlayerScores gets PlayerScore and handles PlayerScore to CBGS.

* Implementation View

*Architecturally significant components*





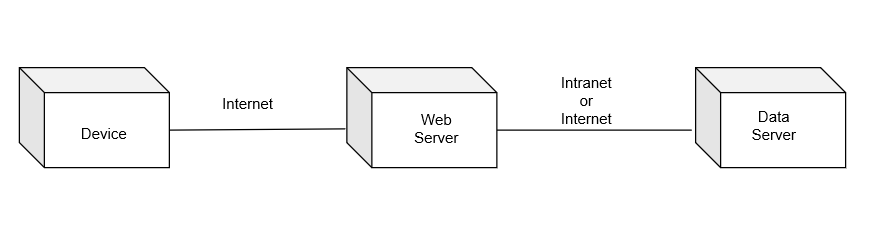


Discussion and Motivation

There is a database component. I have created database ClubBaistGolfSystem. Schema used is dbo. Then there are table components. Each table component is the table that is created such as DailyTeeSheet, Player, TeeTime, StandingTeeTimeRequest, Member and PlayerScore. There is a stored procedure container which contains all the stored procedure required to BookTeeTime, SubmitStandingTeeTimeRequest, RecordsPlayerScores. Below I have all the file component. Each file component is the file I have created. These are the razor pages which contains all the files with .cshtml extension and .cshtml.cs extension and also it contains files with .cs extension that are the classes.

* Deployment View

Architecturally significant nodes and physical configuration between node



* Discussion and Motivation

Device- It is the device that is used by the Player, Clerk, Pro-shop staff, Member, Membership Committee.

Hardware- PC, Mobile, Tablet

Software- Web Browser

* Web Server

Hardware- Web Server

Software- Web Hosting Server (WebBaist)

* Data Server

Hardware- Data Server (DataBaist)

Software- DBMS

For submitting request such as BookTeeTime Player, or Clerk or Proshop staff from his device submits request to the system. For RecordsPlayerScores, player submits the request to the system from his device, for SubmitsStandingTeeTimeRequest ShareholderMember submits request to the system from his device which then passes to the web server through http. Web Server then processes the request and forwards it to the Data Server and the via-versa.