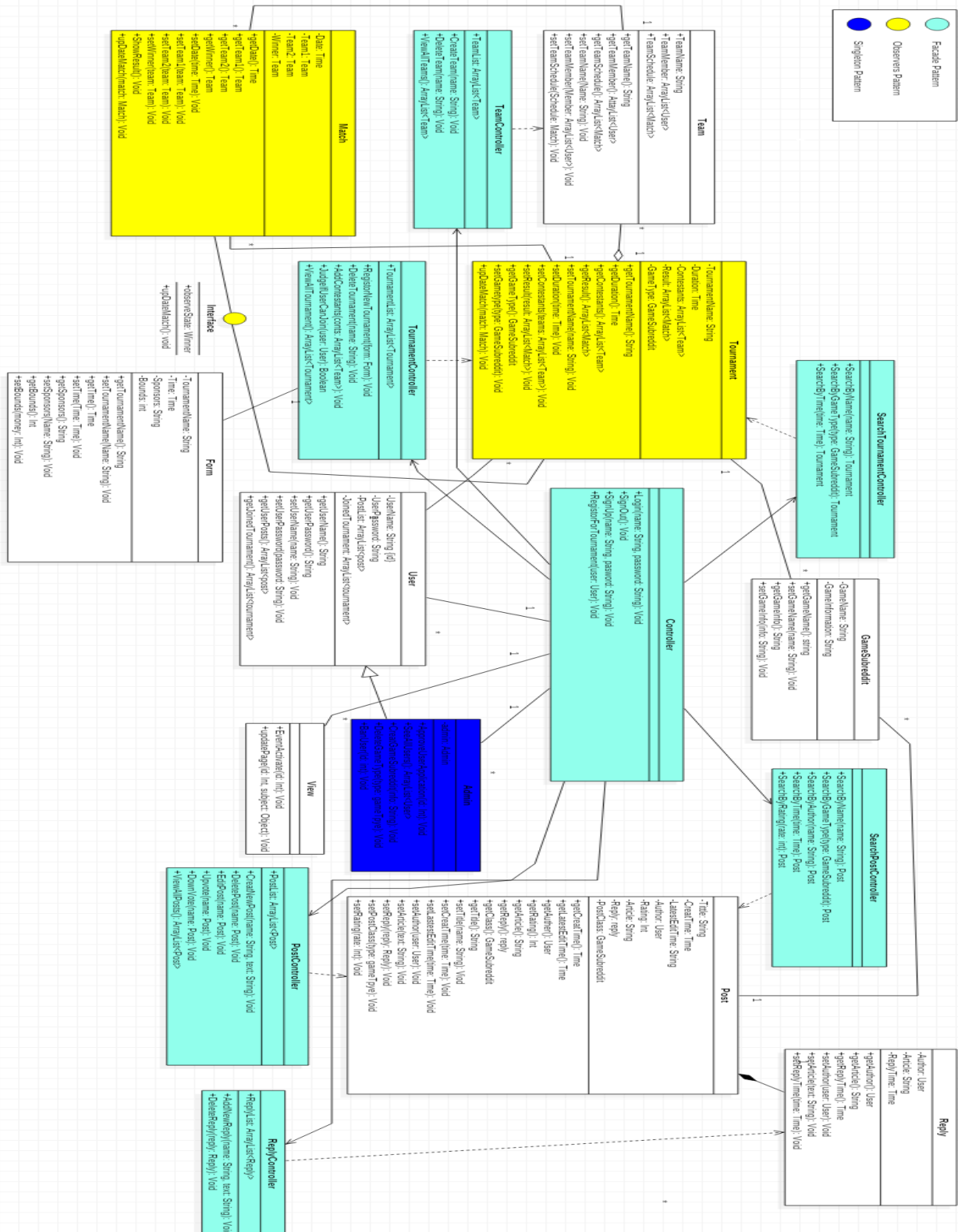**Project Team Name:** Game DashBoard, Team 08
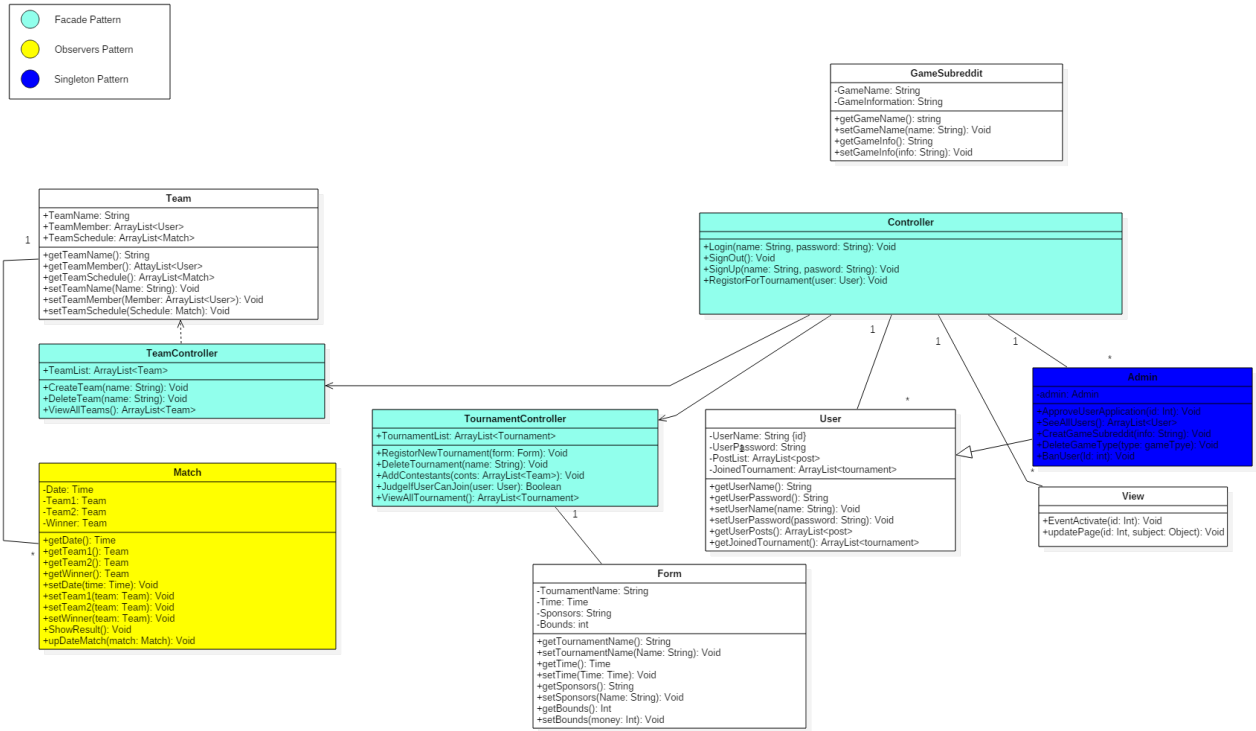**Members**: Yang Yang, Zhaozhong Peng, Yuhou Wang
Vision: Create a place where you can discuss games and compete against each other.
Description: The game dashboard website is where users can browse subboards for discussion/information about games they like and post their own news/discussion. In addition, there will be tournament hosting(bracket style) so users can compete against each other.

Previous Class Diagram:

Current Class Diagram:

Facade Pattern
Observers Pattern
Singleton Pattern

GameSubreddit
-GameName: String
-GameInformation: String
+getGameName(): string
+setGameName(name: String): Void
+getGameInfo(): String
+setGameInfo(info: String): Void

Team
+TeamName: String
+TeamMember: ArrayList<User>
+TeamSchedule: ArrayList<Match>
+getTeamName(): String
+getTeamMember(): AttayList<User>
+getTeamSchedule(): ArrayList<Match>
+setTeamName(Name: String): Void
+setTeamMember(Member: ArrayList<User>): Void
+setTeamSchedule(Schedule: Match): Void

Controller
+Login(name: String, password: String): Void
+SignOut(): Void
+SignUp(name: String, pasword: String): Void
+RegistorForTournament(user: User): Void

TeamController
+TeamList: ArrayList<Team>
+CreateTeam(name: String): Void
+DeleteTeam(name: String): Void
+ViewAllTeams(): ArrayList<Team>

TournamentController
+TournamentList: ArrayList<Tournament>
+RegistorNewTournament(form: Form): Void
+DeleteTournament(name: String): Void
+AddContestants(conts: ArrayList<Team>): Void
+JudgeIfUserCanJoin(user: User): Boolean
+ViewAllTournament(): ArrayList<Tournament>

User
-UserName: String {id}
-UserPassword: String
-PostList: ArrayList<post>
-JoinedTournament: ArrayList<tournament>
+getUserName(): String
+getUserPassword(): String
+setUserName(name: String): Void
+setUserPassword(password: String): Void
+getUserPosts(): ArrayList<post>
+getJoinedTournament(): ArrayList<tournament>

Admin
-admin: Admin
+ApproveUserApplication(id: Int): Void
+SendAllient(): ArrayList of Users
+CreateGameSubreddit(info: String): Void
+DeleteGameType(type: gameType): Void
+BanUser(id: Int): Void

View
+EventActivate(id: Int): Void
+updatePage(id: Int, subject: Object): Void

Match
-Date: Time
-Team1: Team
-Team2: Team
-Winner: Team
+getDate(): Time
+getTeam1(): Team
+getTeam2(): Team
+getWinner(): Team
+setDate(time: Time): Void
+setTeam1(team: Team): Void
+setTeam2(team: Team): Void
+setWinner(team: Team): Void
+ShowResult(): Void
+upDateMatch(match: Match): Void

Form
-TournamentName: String
-Time: Time
-Sponsors: String
-Bounds: int
+getTournamentName(): String
+setTournamentName(Name: String): Void
+getTime(): Time
+setTime(Time: Time): Void
+getSponsors(): String
+setSponsors(Name: String): Void
+getBounds(): Int
+setBounds(money: Int): Void

Summary:

For the last two weeks, we have decided to use Spring Framework instead of Django. We have started learning Spring Framework and started implementing our Tournament design first and leave the Forums alone for now. We have started making use of Inversion of Control Design Pattern and Model View Controller Design PAttern

BreakDown:

All: studied Spring Framework
 Zhaozhong Peng: Updated previous Class Diagram. wrote Tournament Controller and form
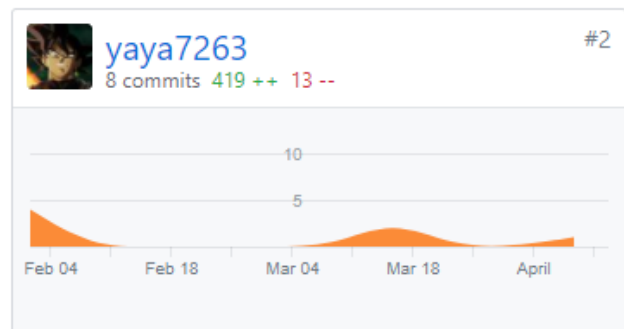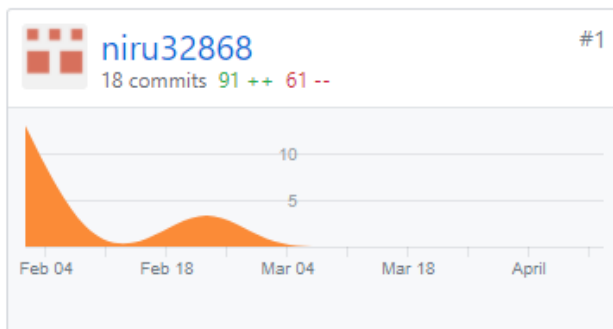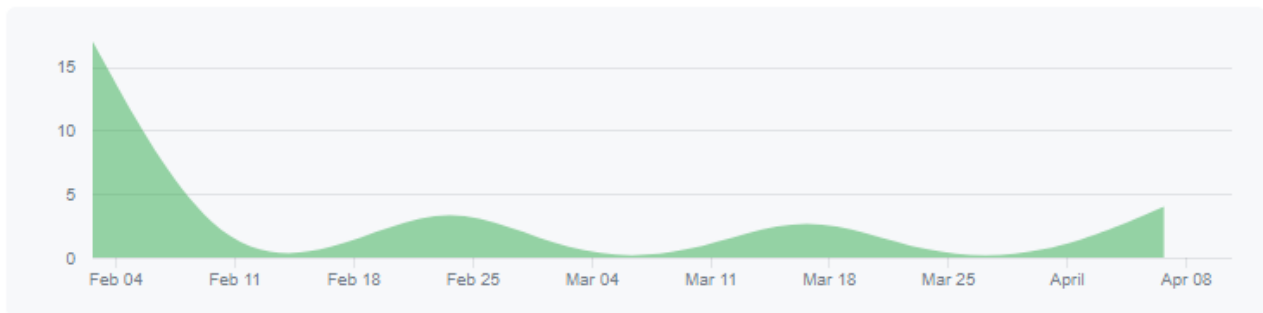Yuhou Wang: Wrote Match and TeamController
Yang Yang: Wrote Beans.xml, Game.java, MainApp.java

GitHub Graph:

Feb 4, 2018 – Apr 12, 2018

Contributions to master, excluding merge commits

niru32868 #1
18 commits 91 ++ 61 --

yaya7263 #2
8 commits 419 ++ 13 --

pengsrif #3
3 commits 109 ++ 0 --

wanghou5 #4
1 commit 57 ++ 0 --

Estimate Remaining Effort:
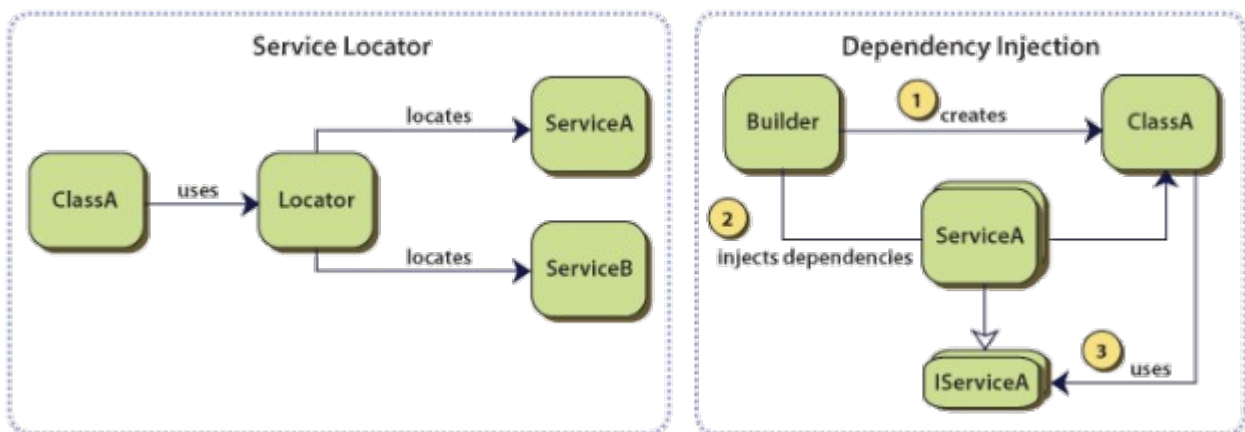
Our team leader has left us. Alas, I knew him well.

We have a lot of work left to be done. Our seperate files have yet to be tied together.
We still have some Spring Framework learning to be done, like Spring boot.

We will finish learning Spring, and modify our MvC so that it works with our Controller.

Design Pattern:

We have applied Inversion of Control design pattern with Dependency Injection.
We have created a Beans.Xml that supplies the dependencies of Game(GameSubreddit).

9. Inversion of Control Design Pattern

Final Iteration:

 Our final iteration will probably only contain the Tournament. We do not think we have time to implement the game reddit as well. We are planning to use Spring boot and make a video so we can better present our project.