AfterQuestionScorePageView

```
- model: com.god.kahit.viewModel.AfterQuestionScorePageViewModel
- animator: ObjectAnimator
- recyclerView: RecyclerView
- recyclerAdapter: RecyclerView.Adapter
- layoutManager: RecyclerView.LayoutManager

# onCreate(Bundle): void
- setupRecycler(): void
+ launchQuestionView(): void
+ launchCategoryView(): void
- startTimer(ProgressBar): void
```

CustomSpinnerAdapter

```
- mColors: List<Integer>

+ getDropDownView(int , View , ViewGroup ): View
```

HotSwapAddPlayersView

```
- LOG_TAG: String {read only}
- recyclerView: RecyclerView
- recyclerAdapter: RecyclerView.Adapter
~ layoutManager: RecyclerView.LayoutManager
~ playerMap: MutableLiveData<List<Pair<Player, Integer>>>
~ hotSwapAddPlayersViewModel: HotSwapAddPlayersViewModel

# onCreate(Bundle): void
- setupRecyclerView(): void
+ onPlayerClick(int): void
+ onTeamSelected(int, int): void
# onDestroy(): void
+ launchBackHotSwapGameModeView(View ): void
+ launchQuestionView(View): void
```

JoinRoomView

```
- LOG_TAG: String {read only}
~ list: MutableLiveData<List<String>>
~ joinRoomViewModel: JoinRoomViewModel

# onCreate(Bundle): void
+ launchBackChooseGameView(View): void
+ launchTeamArrangementActivity(View): void
```

LotteryView

```
~ lotteryViewModel: LotteryViewModel
~ constraintLayout: ConstraintLayout
~ textViewList: List<TextView>
~ imageViewList: List<ImageView>
~ playerImageViews: List<ImageView>
~ playerNameTxtViews: List<TextView>
- numOfPlayers: int
- count: int
- maxCount: int
- playerMap: MutableLiveData<Map<Integer, String>>
- lotteryItemMap: MutableLiveData<Map<Integer, Item>>

# onCreate(Bundle): void
- initLottery(): void
- initLiveData(): void
- populateLayoutViewDynamically(): void
+ setUpImageViewList(int, int, int): void
- setUpTextViewList(int, int, int): void
- getCenterChildId(): int
- setupPlayerTextViews(): List<TextView>
- setupPlayerImageViews(): List<ImageView>
- incCounter(): void
- isDone(): boolean
+ getImageId(int): int
+ displayLottery(): void
+ getPlayerImageItem(int): int
+ imageAnimation(int): void
```

StoreView

```
- storeViewModel: StoreViewModel
- pointsText: TextView
- itemType1: TextView
- itemType2: TextView
- itemType3: TextView
- itemsIcons: ArrayList<ImageView>
- drawerLayout: DrawerLayout
- storeImage: ImageView
- itemButtons: List<Button>

# onCreate(Bundle): void
+ initializeStoreView(): void
+ populateItemIcons(): void
+ findViews(): void
+ findItemIcons(): void
+ findItemButtons(): void
+ addDrawerListener(): void
+ setPointsText(): void
+ setItemTypes(): void
+ addActionsToButtons(): void
+ buy(Button itemButton): void
+ disablebuttons(): void
+ setButtonText(): void
```

CategoryView

```
~ Buttons: List<ImageButton>
~ model: CategoryViewModel

# onCreate(Bundle): void
- addPicturesToButton(): void
+ onCategoryClick(View): void
+ launchQuestionView(): void
```

HotSwapGameModeView

```
- LOG_TAG: String {read only}
~ hotSwapGameModeViewModel: HotSwapGameModeViewModel
- gameModes: MutableLiveData<List<String>>

# onCreate(Bundle): void
+ launchHotSwapAddPlayerView(View): void
+ launchBackChooseGameView(View): void
```

HotSwapRecyclerAdapter

```
- LOG_TAG: String {read only}
- TYPE_ITEM: int= 0 {read only}
- TYPE_FOOTER: int= 1 {read only}
- TYPE_DIVIDER: int= 2 {read only}
- iOnplayerclickListener: IOnPlayerClickListener
~ playerList: MutableLiveData<List<Pair<Player, Integer>>>
~ teamColors: List<Integer>
~ teamNumbers: List<String>
- context: Context
```

itemViewHolder

```
~ iOnplayerclickListener: IOnPlayerClickListener
+ row: ConstraintLayout
+ textView: TextView
+ img ImageView
+ remove: Button
+ spin: Spinner

+ onItemSelected(AdapterView<?>, View, int, long): void
+ onNothingSelected(AdapterView<?>): void
+ onClick(View): void
```

MainActivityView

```
- LOG_TAG: String {read only}

# onCreate(Bundle): void
+ launchChooseGameClass(View): void
+ launchHowToPlayView(View): void
+ launchPreGameCoundown(View): void
+ launchSettingsView(View): void
```

view

QuestionView

```
- LOG_TAG: String {read only}
- h1: Handler {read only}
~ qTime: int
~ n: int
~ k: int
~ p1: String
- animation: ObjectAnimator
- answers: ArrayList<TextView>
- model: QuestionViewModel

# onCreate(Bundle): void
+ initAnswerTextViews(): void
+ launchScorePageClass(): void
+ launchAfterQuestionScorePageClass(): void
+ launchBackMainActivityClass(View): void
+ onBackPressed(): void
+ OnAnswerClicked(View): void
+ populateQuestionTextView(String): void
+ populateAnswerTextViews(List<String>): void
+ populatePlayerName(String): void
+ populateQuestionNum(int): void
+ populateTotalNumQuestions(int): void
+ startTimer(ProgressBar, final int): void
```

TeamArrangementRecyclerAdapter

```
- LOG_TAG: String {read only}
- TYPE_ITEM: int {read only}
- TYPE_FOOTER: int {read only}
- iOnplayerclickListener;: IOnPlayerClickListener
~ playerList: MutableLiveData<List<Player>>
~ teamColors: List<Integer>
~ teamNumbers: List<String>
- context: Context
```

itemViewHolder

```
~ iOnplayerclickListener: IOnPlayerClickListener
+ row: ConstraintLayout
+ textView: TextView
+ img ImageView
+ remove: Button
+ spin: Spinner

+ onItemSelected(AdapterView<?>, View, int, long): void
+ onNothingSelected(AdapterView<?>): void
+ onClick(View): void
```

ChooseGameView

```
- LOG_TAG: String {read only}

# onCreate(Bundle): void
+ launchLotteryView(View): void
+ launchQuestionView(View): void
+ ChooseGameView(View): void
+ launchJoinRoomView(View): void
+ launchHotSwapGameModeView(View): void
+ launchSidenavTest(View): void
```

HostCreateRoomView

```
- LOG_TAG: String {read only}
~ list: MutableLiveData<List<String>>
~ hostCreateRoomViewModel: HostCreateRoomViewModel

# onCreate(Bundle): void
+ launchBackChooseGameView(View): void
+ launchQuestionView(View): void
+ launchTeamArrangementView(View): void
```

HowToPlayView

```
# onCreate(Bundle): void
```

<<IOnPlayerClickListener>>

```
onPlayerClick(int): void
onTeamSelected(int, int): void
```

FooterViewHolder

```
~ btnSubmitProblem: Button

+ onClick(View): void
- initTeamNumbers(): void
- initTeamColors(): void
+ onCreateViewHolder(ViewGroup, int): itemViewHolder
+ onBindViewHolder(itemViewHolder, int): void
+ getItemViewType(int): int
+ getItemCount(): int
```

PreGameCountdownView

```
~ preGameCountdownTimer: preGameCountdownViewModel
~ text: TextView
~ counter: CountDownTimer

# onCreate(Bundle): void
- startTimer(): void
```

ScorePageAdapter

```
- playerScoreDeltaList: List <Tuple<String, String>>

+ onCreateViewHolder(ViewGroup, int): ScorePageAdapter.ViewHolder
+ onBindViewHolder(ViewHolder, int): void
+ getItemCount(): int
```

ViewHolder

```
+ name: TextView
+ score: TextView
+ img: ImageView
```

ScorePageClass

```
- LOG_TAG: String {read only}

# onCreate(Bundle): void
+ launchBackMainActivityClass(View): void
+ onBackPressed(): void
```

SettingsView

```
~ musicSwitch: Switch

# onCreate(Bundle): void
- switchCheckListener(): void
```

TeamArrangementView

```
- LOG_TAG: String {read only}
- recyclerView: RecyclerView
- recyclerAdapter: RecyclerView.Adapter
~ layoutManager: RecyclerView.LayoutManager
~ playerList: MutableLiveData<List<Pair<Player, Integer>>>
~ teamArrangementViewModel: TeamArrangementViewModel

# onCreate(Bundle): void
- setupRecyclerView(): void
# onDestroy(): void
+ onPlayerClick(int): void
+ onTeamSelected(int, int): void
```

FooterViewHolder

```
~ btnSubmitProblem: Button

+ onClick(View): void
- initTeamNumbers(): void
- initTeamColors(): void
+ onCreateViewHolder(ViewGroup, int): itemViewHolder
+ onBindViewHolder(itemViewHolder, int): void
+ getItemViewType(int): int
+ getItemCount(): int
```