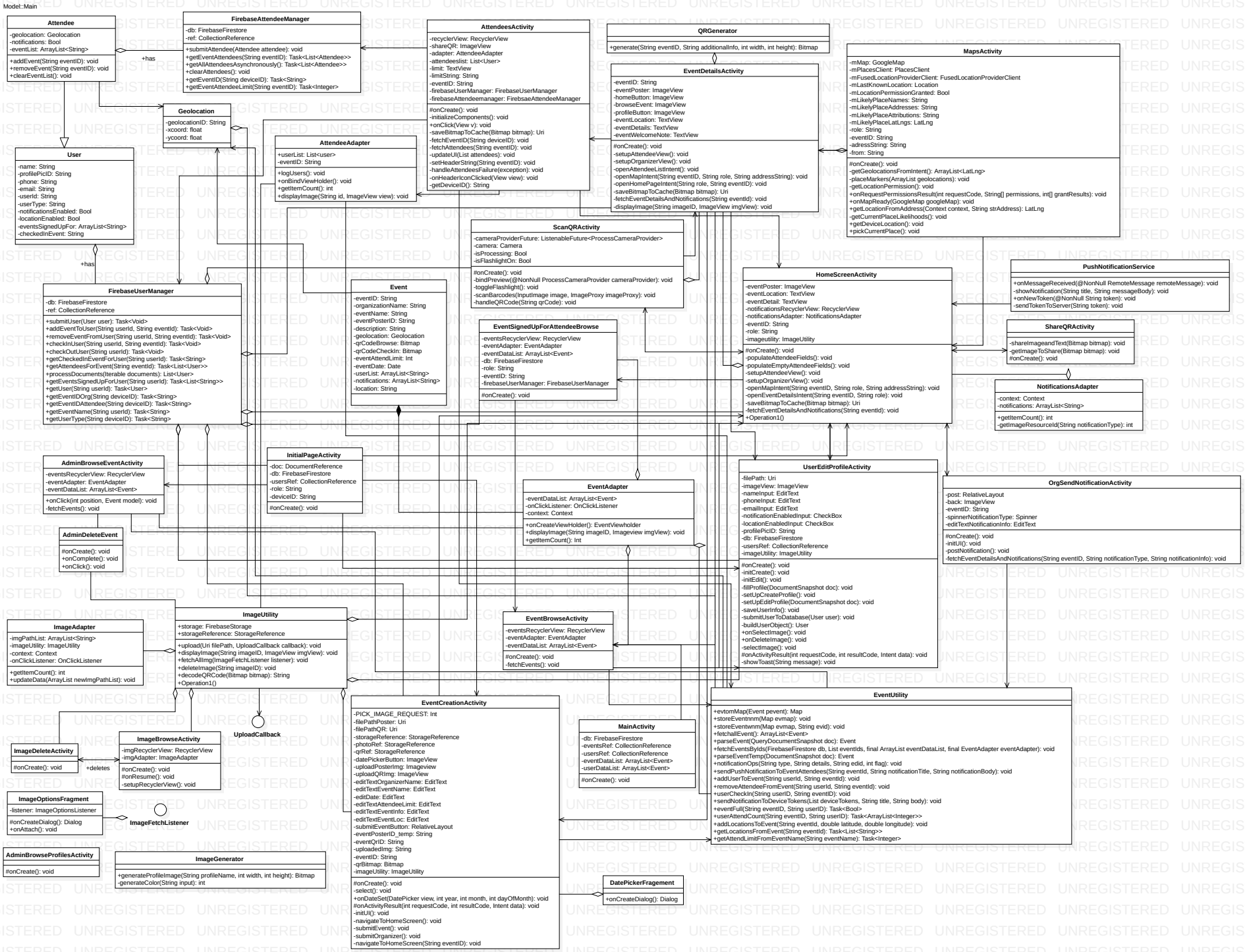
Model::Main


Attendee
-geolocation: Geolocation
-notifications: Bool
-eventList: ArrayList<String>
+addEvent(String eventID): void
+removeEvent(String eventID): void
+clearEventList(): void
+has
FirebaseAttendeeManager
-db: FirebaseFirestore
-ref: CollectionReference
+submitAttendee(Attendee attendee): void
+getEventAttendees(String eventID): Task<List<Attendee>>
+getAllAttendeesAsynchronously(): Task<List<Attendee>>
+clearAttendees(): void
+getEventID(String deviceID): Task<String>
+getEventAttendeeLimit(String eventID): Task<Integer>
AttendeesActivity
-recyclerView: RecyclerView
-shareQR: ImageView
-adapter: AttendeeAdapter
-attendeeslist: List<User>
-limit: TextView
-limitString: String
-eventID: String
-firebaseUserManager: FirebaseUserManager
-firebaseAttendeemanager: FirebsaeAttendeeManager
#onCreate(): void
-initializeComponents(): void
+onClick(View v): void
-saveBitmapToCache(Bitmap bitmap): Uri
-fetchEventID(String deviceID): void
-fetchAttendees(String eventID): void
-updateUI(List attendees): void
-setHeaderString(String eventID): void
-handleAttendeesFailure(exception): void
-onHeaderIconClicked(View view): void
-getDeviceID(): String
QRGenerator
+generate(String eventID, String additionalInfo, int width, int height): Bitmap
MapsActivity
-mMap: GoogleMap
-mPlacesClient: PlacesClient
-mFusedLocationProviderClient: FusedLocationProviderClient
-mLastKnownLocation: Location
-mLocationPermissionGranted: Bool
-mLikelyPlaceNames: String
-mLikelyPlaceAddresses: String
-mLikelyPlaceAttributions: String
-mLikelyPlaceLatLngs: LatLng
-role: String
-eventID: String
-adressString: String
-from: String
#onCreate(): void
-getGeolocationsFromIntent(): ArrayList<LatLng>
-placeMarkers(ArrayList geolocations): void
-getLocationPermission(): void
+onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults): void
+onMapReady(GoogleMap googleMap): void
+getLocationFromAddress(Context context, String strAddress): LatLng
-getCurrentPlaceLikelihoods(): void
+getDeviceLocation(): void
+pickCurrentPlace(): void
EventDetailsActivity
-eventID: String
-eventPoster: ImageView
-homeButton: ImageView
-browseEvent: ImageView
-profileButton: ImageView
-eventLocation: TextView
-eventDetails: TextView
-eventWelcomeNote: TextView
#onCreate(): void
-setupAttendeeView(): void
-setupOrganizerView(): void
-openAttendeeListIntent(): void
-openMapIntent(String eventID, String role, String addressString): void
-openHomePageIntent(String role, String eventID): void
-saveBitmapToCache(Bitmap bitmap): Uri
-fetchEventDetailsAndNotifications(String eventId): void
-displayImage(String imageID, ImageView imgView): void
Geolocation
-geolocationID: String
-xcoord: float
-ycoord: float
AttendeeAdapter
+userList: List<user>
-eventID: String
+logUsers(): void
+onBindViewHolder(): void
+getItemCount(): int
+displayImage(String id, ImageView view): void
User
-name: String
-profilePicID: String
-phone: String
-email: String
-userId: String
-userType: String
-notificationsEnabled: Bool
-locationEnabled: Bool
-eventsSignedUpFor: ArrayList<String>
-checkedInEvent: String
+has
ScanQRActivity
-cameraProviderFuture: ListenableFuture<ProcessCameraProvider>
-camera: Camera
-isProcessing: Bool
-isFlashlightOn: Bool
#onCreate(): void
-bindPreview(@NonNull ProcessCameraProvider cameraProvider): void
-toggleFlashlight(): void
-scanBarcodes(InputImage image, ImageProxy imageProxy): void
-handleQRCode(String qrCode): void
HomeScreenActivity
-eventPoster: ImageView
-eventLocation: TextView
-eventDetail: TextView
-notificationsRecyclerView: RecyclerView
-notificationsAdapter: NotificationsAdapter
-eventID: String
-role: String
-imageutility: ImageUtility
#onCreate(): void
-populateAttendeeFields(): void
-populateEmptyAttendeeFields(): void
-setupAttendeeView(): void
-setupOrganizerView(): void
-openMapIntent(String eventID, String role, String addressString): void
-openEventDetailsIntent(String eventID, String role): void
-saveBitmapToCache(Bitmap bitmap): Uri
-fetchEventDetailsAndNotifications(String eventId): void
+Operation1()
PushNotificationService
+onMessageReceived(@NonNull RemoteMessage remoteMessage): void
-showNotification(String title, String messageBody): void
+onNewToken(@NonNull String token): void
-sendTokenToServer(String token): void
FirebaseUserManager
-db: FirebaseFirestore
-ref: CollectionReference
+submitUser(User user): Task<Void>
+addEventToUser(String userId, String eventId): Task<Void>
+removeEventFromUser(String userId, String eventId): Task<Void>
+checkInUser(String userId, String eventId): Task<Void>
+checkOutUser(String userId): Task<Void>
+getCheckedInEventForUser(String userId): Task<String>
+getAttendeesForEvent(String eventId): Task<List<User>>
+processDocuments(Iterable documents): List<User>
+getEventsSignedUpForUser(String userId): Task<List<String>>
+getUser(String userId): Task<User>
+getEventIDOrg(String deviceID): Task<String>
+getEventIDAttendee(String deviceID): Task<String>
+getEventName(String userId): Task<String>
+getUserType(String deviceID): Task<String>
Event
-eventID: String
-organizationName: String
-eventName: String
-eventPosterID: String
-description: String
-geolocation: Geolocation
-qrCodeBrowse: Bitmap
-qrCodeCheckIn: Bitmap
-eventAttendLimit: Int
-eventDate: Date
-userList: ArrayList<String>
-notifications: ArrayList<String>
-location: String
EventSignedUpForAttendeeBrowse
-eventsRecyclerView: RecyclerView
-eventAdapter: EventAdapter
-eventDataList: ArrayList<Event>
-db: FirebaseFirestore
-role: String
-eventID: String
-firebaseUserManager: FirebaseUserManager
#onCreate(): void
ShareQRActivity
-shareImageandText(Bitmap bitmap): void
-getImageToShare(Bitmap bitmap): void
#onCreate(): void
NotificationsAdapter
-context: Context
-notifications: ArrayList<String>
+getItemCount(): int
-getImageResourceId(String notificationType): int
AdminBrowseEventActivity
-eventsRecyclerView: RecyclerView
-eventAdapter: EventAdapter
-eventDataList: ArrayList<Event>
+onClick(int position, Event model): void
-fetchEvents(): void
InitialPageActivity
-doc: DocumentReference
-db: FirebaseFirestore
-usersRef: CollectionReference
-role: String
-deviceID: String
#onCreate(): void
EventAdapter
-eventDataList: ArrayList<Event>
-onClickListener: OnClickListener
-context: Context
+onCreateViewHolder(): EventViewholder
+displayImage(String imageID, Imageview imgView): void
+getItemCount(): Int
UserEditProfileActivity
-filePath: Uri
-imageView: ImageView
-nameInput: EditText
-phoneInput: EditText
-emailInput: EditText
-notificationEnabledInput: CheckBox
-locationEnabledInput: CheckBox
-profilePicID: String
-db: FirebaseFirestore
-usersRef: CollectionReference
-imageUtility: ImageUtility
#onCreate(): void
-initCreate(): void
-initEdit(): void
-fillProfile(DocumentSnapshot doc): void
-setUpCreateProfile(): void
-setUpEditProfile(DocumentSnapshot doc): void
-saveUserInfo(): void
-submitUserToDatabase(User user): void
-buildUserObject(): User
+onSelectImage(): void
+onDeleteImage(): void
-selectImage(): void
#onActivityResult(int requestCode, int resultCode, Intent data): void
-showToast(String message): void
OrgSendNotificationActivity
-post: RelativeLayout
-back: ImageView
-eventID: String
-spinnerNotificationType: Spinner
-editTextNotificationInfo: EditText
#onCreate(): void
-initUI(): void
-postNotification(): void
-fetchEventDetailsAndNotifications(String eventID, String notificationType, String notificationInfo): void
AdminDeleteEvent
#onCreate(): void
+onComplete(): void
+onClick(): void
ImageAdapter
-imgPathList: ArrayList<String>
-imageUtility: ImageUtility
-context: Context
-onClickListener: OnClickListener
+getItemCount(): int
+updateData(ArrayList newImgPathList): void
ImageUtility
+storage: FirebaseStorage
+storageReference: StorageReference
+upload(Uri filePath, UploadCallback callback): void
+displayImage(String imageID, ImageView imgView): void
+fetchAllImg(ImageFetchListener listener): void
+deleteImage(String imageID): void
+decodeQRCode(Bitmap bitmap): String
+Operation1()
EventBrowseActivity
-eventsRecyclerView: RecyclerView
-eventAdapter: EventAdapter
-eventDataList: ArrayList<Event>
#onCreate(): void
-fetchEvents(): void
UploadCallback
ImageDeleteActivity
#onCreate(): void
+deletes
ImageBrowseActivity
-imgRecyclerView: RecyclerView
-imgAdapter: ImageAdapter
#onCreate(): void
#onResume(): void
-setupRecyclerView(): void
EventCreationActivity
-PICK_IMAGE_REQUEST: Int
-filePathPoster: Uri
-filePathQR: Uri
-storageReference: StorageReference
-photoRef: StorageReference
-qrRef: StorageReference
-datePickerButton: ImageView
-uploadPosterImg: Imageview
-uploadQRImg: ImageView
-editTextOrganizerName: EditText
-editTextEventName: EditText
-editDate: EditText
-editTextAttendeeLimit: EditText
-editTextEventInfo: EditText
-editTextEventLoc: EditText
-submitEventButton: RelativeLayout
-eventPosterID_temp: String
-eventQrID: String
-uploadedImg: String
-eventID: String
-qrBitmap: Bitmap
-imageUtility: ImageUtility
#onCreate(): void
-select(): void
+onDateSet(DatePicker view, int year, int month, int dayOfMonth): void
#onActivityResult(int requestCode, int resultCode, Intent data): void
-initUI(): void
-navigateToHomeScreen(): void
-submitEvent(): void
-submitOrganizer(): void
-navigateToHomeScreen(String eventID): void
MainActivity
-db: FirebaseFirestore
-eventsRef: CollectionReference
-usersRef: CollectionReference
-eventDataList: ArrayList<Event>
-userDataList: ArrayList<Event>
#onCreate(): void
EventUtility
+evtomMap(Event pevent): Map
+storeEventnmm(Map evmap): void
+storeEventwnm(Map evmap, String evid): void
+fetchallEvent(): ArrayList<Event>
+parseEvent(QueryDocumentSnapshot doc): Event
+fetchEventsByIds(FirebaseFirestore db, List eventIds, final ArrayList eventDataList, final EventAdapter eventAdapter): void
+parseEventTemp(DocumentSnapshot doc): Event
+notificationOps(String type, String details, String edid, int flag): void
+sendPushNotificationToEventAttendees(String eventId, String notificationTitle, String notificationBody): void
+addUserToEvent(String userId, String eventId): void
+removeAttendeeFromEvent(String userId, String eventId): void
+userCheckIn(String userID, String eventID): void
+sendNotificationToDeviceTokens(List deviceTokens, String title, String body): void
+eventFull(String eventID, String userID): Task<Bool>
+userAttendCount(String eventID, String userID): Task<ArrayList<Integer>>
+addLocationsToEvent(String eventId, double latitude, double longitude): void
+getLocationsFromEvent(String eventId): Task<List<String>>
+getAttendLimitFromEventName(String eventName): Task<Integer>
ImageOptionsFragment
-listener: ImageOptionsListener
#onCreateDialog(): Dialog
+onAttach(): void
ImageFetchListener
AdminBrowseProfilesActivity
#onCreate(): void
ImageGenerator
+generateProfileImage(String profileName, int width, int height): Bitmap
-generateColor(String input): int
DatePickerFragement
+onCreateDialog(): Dialog

Model::EventSignedUpForAttendeeBrowse::Activity1::ActivityDiagram1

UNREGISTERED