Cache

- ourInstance: Cache
- requesters: ArrayList<User>
- habitTypes: ArrayList<HabitType>
- followings: ArraryList<String>
- adapter: CommunityAdapter
- init: boolean

+ getInstance(): Cahe
- Cache
- fakeTestData(): void
- checkUpdates(): boolean

+ getAdapter(Activity): CommunityAdapter

RemoveTask

- activity: Activity

+ RemoveTask(Activity)
# doInBackground(String)
# onPostExecute(Integer): void

FetchTask

---

doInBackground(Void): void

LoadBytesImageTask

- imageView: ImageView
- height: int
- width: int
- image: Bitmap

---

doInBackground(Void... params): vo
onPostExecute(Void aVoid): void

- mCurre
- getImag

- Encode
+ ImageI
+ dispat
- createI
+ setPic(
+ encode
+ encode
+ getPict

AppendTask

- type: String

+ AppendTask(String type)
+ doInBackground(String... strings)

CheckExistTask

---

doInBackground(String... strings): Intege

AsyncTask

- he
- wi
- im
- ur

- lo
- en
d
o

ESManager

- ourInstance: ESManager
- client: JestDroidClient

+ create(Item item): static boolean
+ get(String id, String type, Class cls): Obje
+ update(Item item): static boolean
+ delete(Item item): static boolean

SaveLocalDataTask

---

doInBackground(): void

- habitList: Ha
- buffer: Buffe
- eventList: E

Fragment

ImageManager

...entPhotoPath: String
...geAble: GetImageAble

...eImageTask
...Manager(GetImageAble)
...chTakePictureIntent(Context): Void
...mageFile(Context): File
...ImageView): void
...e(ImageView, Uri): void
...e(ImageView): void
...tureFromGalleryIntent(Context): void

EncodeImageTask

...eight: int
...idth: int
...mageView: ImageView
...i: Uri

...adImage(): Bitmap
...ncodeImage(Bitmap image): void
...oInBackground(Void... params): void
...nPostExecute(Void aVoid): void

FragmentProfile

- imageManager: ImageManager

+ newInstance(): FragmentProfile
+ onCreateView(LayoutInflater, ViewGroup, Bundle): View
+ refresh(): void
+ result(): void

FragmentHabitList

- adapter: HabitListAdapter

+ newInstance(Context): FraagmentHabitList
+ onCreateView(LayoutInflater, ViewGroup, Budle): Vie...
+ onStart(): void

- list...
- se...
- us...

+ ne...
+ on...

FragmentEventList

- adapter: EventListAdapter
- keyword: String
- showTypes: HashMap<String, Boolean>

+ newInstance(): FragmentEventList
+ onCreateView(LayoutInflater, ViewGroup, Bunde): Vie...
+ onQueryTextSubmit(String): boolean
+ onQueryTextChange(String): boolean
+ onStart(): void

DataManager

...abitList
...er
...ventList

ActivityModifyEvent

ActivityMain

- viewPager: ViewPager
- meunItem: MenuItem
- bottomNavigationView: BottomNavigationView
- that: ActivityMain

FragmentCommunity

:ListView
archBox: EditText
erToDisplay:ArrayList<user>

---

ewInstance(String): FragmentEventList
nCreateView(LayoutInflater, ViewGroup, Bundle): View

1

ActivityHabitMap
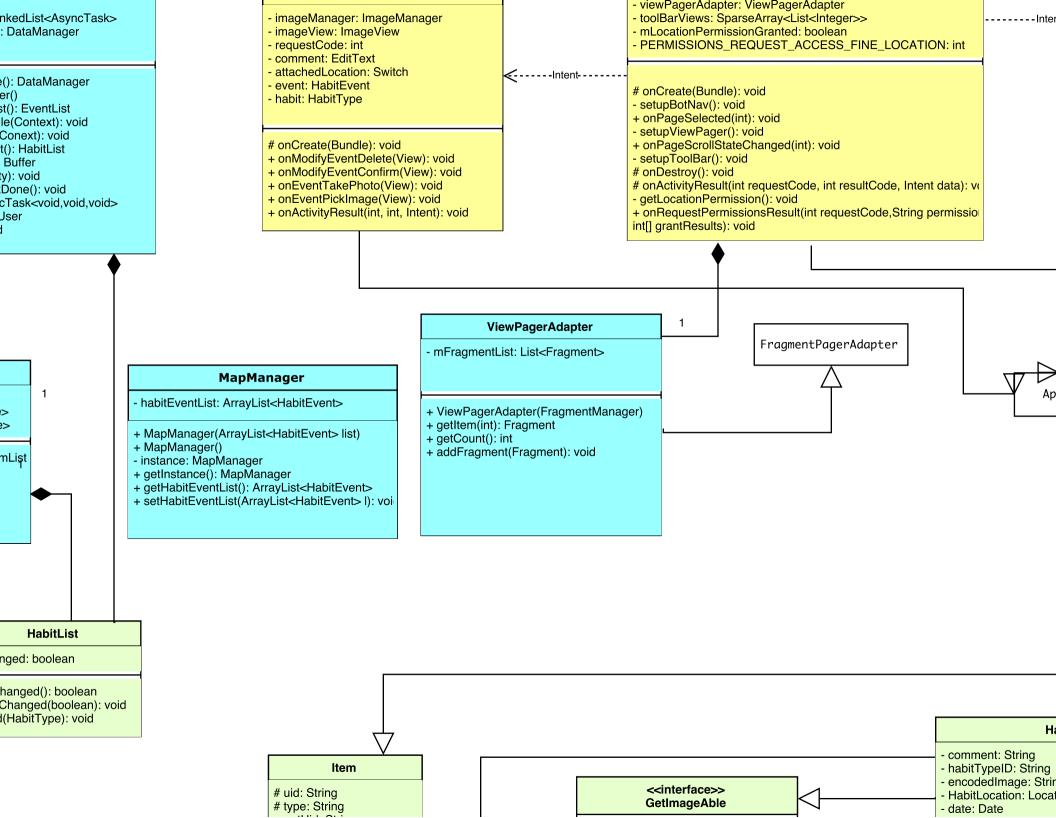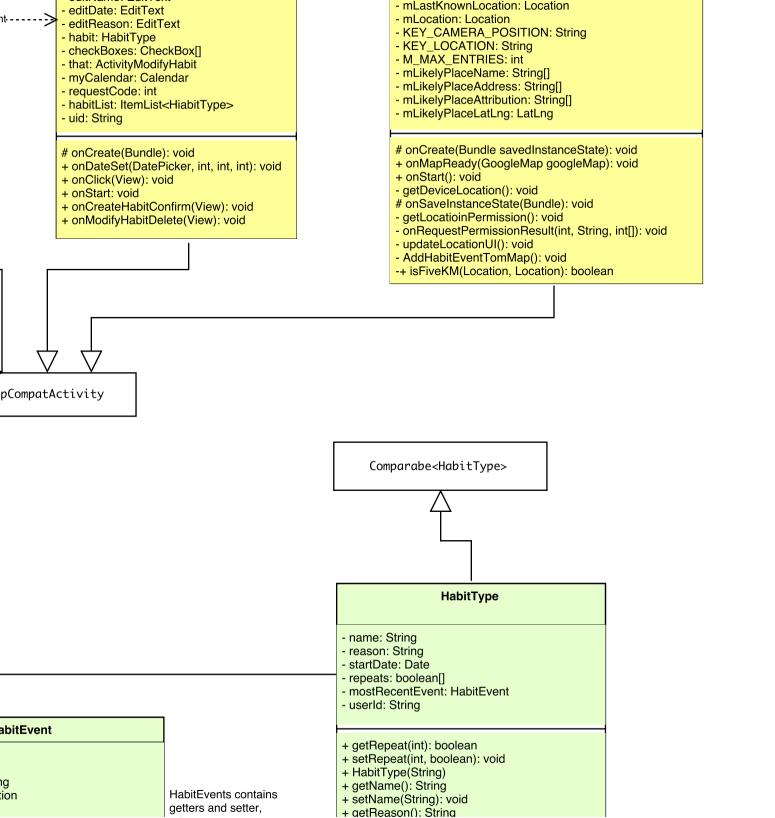
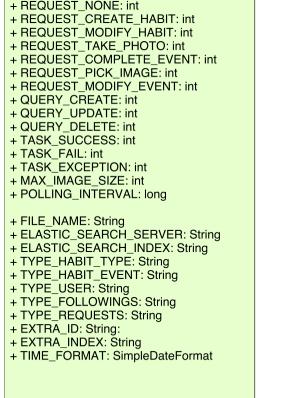- mMap: GoogleMap
- mgeoDataClinet: GeoDataClinet
- mCameraPosition: CameraPosition
- mFusedLocationProviderClient: FusedLocationProvider(
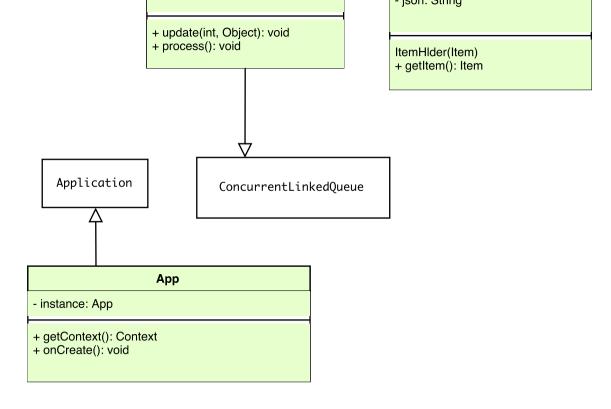- mDefaultLocation: LatLng
- DEFAULT_ZOOM: int
- PERMISSIONS_REQUEST_ACCESS_FINE_LOCATIO
- mLocationPermissionGranted: boolean

ActivityModifyHabit

- editName: EditText

+ delete(Item Item): static boolean
- ESManager()

CancelAccountTask

doInBackground(Void... voids): void

- user:User
- taskPool: Li
- ourInstance

+ getInstance
- DataManag
+ getEventLis
+ loadFromFi
+ saveInFile(
+ getHabitLis
+ getBuffer():
+ login(Activit
+ waitAllTask
+ save: Asyn
+ getUser(): U
+ clear(): void

EventList

+ add(HabitEvent): void
+ remove(Object): HabitEvent
+ commit(UUID): void

BaseAdapter

EventListAdapter

- activity: Activity
- modelList: ItemList<HabitEvent>
- eventList: ArrayList<HabitEvent>

+ EventListAdapter: (Activity, EventList)
+ getCount(): Int
+ commit(HashMap<String,Boolean> showTypes,
        String keyword)
+ getView(int, View, ViewGroup)
+ getItemId(int): long
+ getItem(int): Object

CommunityAdapter

- requestList: ArrayList<Use
- habitTypes: ArrayList<Hab
- activity: Activity

+getCount(): int
+getItem(): Object
+getItemId(): long
+getView: view
+commit(): void

HabitListAdapter

- context: Context
- habitList: ArrayList<HabitType
- modelList: ItemList<HabitType

+ HabitListAdapter(Context, Iter
+ commit(): void
+ getItem(int): Object
+ getitemId(int): long
+ getCount(): int
+ getView(int): View

ListAdapter

ItemList

+ add(T): void
+ remove(Object): T
+ commit(String): void

- cha

+ isC
+ set
+ add

HashMap<String>

<<interface>>
Constant

Constant contians
many public constant

Buffer

ItemHolder

- map: HashMap<String,Class>
- type: String

(Partial class - top left, cyan)

```
nkedList<AsyncTask>
: DataManager
```

```
e(): DataManager
er()
st(): EventList
le(Context): void
Conext): void
t(): HabitList
 Buffer
ty): void
Done(): void
cTask<void,void,void>
User
d
```

(Class - top center, yellow)

```
- imageManager: ImageManager
- imageView: ImageView
- requestCode: int
- comment: EditText
- attachedLocation: Switch
- event: HabitEvent
- habit: HabitType
```

```
# onCreate(Bundle): void
+ onModifyEventDelete(View): void
+ onModifyEventConfirm(View): void
+ onEventTakePhoto(View): void
+ onEventPickImage(View): void
+ onActivityResult(int, int, Intent): void
```

(Class - top right, yellow)

```
- viewPagerAdapter: ViewPagerAdapter
- toolBarViews: SparseArray<List<Integer>>
- mLocationPermissionGranted: boolean
- PERMISSIONS_REQUEST_ACCESS_FINE_LOCATION: int
```

```
# onCreate(Bundle): void
- setupBotNav(): void
+ onPageSelected(int): void
- setupViewPager(): void
+ onPageScrollStateChanged(int): void
- setupToolBar(): void
# onDestroy(): void
# onActivityResult(int requestCode, int resultCode, Intent data): v
- getLocationPermission(): void
+ onRequestPermissionsResult(int requestCode,String permission
int[] grantResults): void
```

MapManager (cyan)

```
- habitEventList: ArrayList<HabitEvent>
```

```
+ MapManager(ArrayList<HabitEvent> list)
+ MapManager()
- instance: MapManager
+ getInstance(): MapManager
+ getHabitEventList(): ArrayList<HabitEvent>
+ setHabitEventList(ArrayList<HabitEvent> l): voi
```

ViewPagerAdapter (cyan)

```
- mFragmentList: List<Fragment>
```

```
+ ViewPagerAdapter(FragmentManager)
+ getItem(int): Fragment
+ getCount(): int
+ addFragment(Fragment): void
```

FragmentPagerAdapter

```
Ap
```

HabitList (green)

```
nged: boolean
```

```
hanged(): boolean
Changed(boolean): void
d(HabitType): void
```

Item (green)

```
# uid: String
# type: String
```

<<interface>> GetImageAble (green)

Ha (green)

```
- comment: String
- habitTypeID: String
- encodedImage: Strin
- HabitLocation: Locat
- date: Date
```

Connector labels: Intent, 1

Partial UML class diagram (text content):

Top-left class (yellow):
- editName: EditText
- editDate: EditText
- editReason: EditText
- habit: HabitType
- checkBoxes: CheckBox[]
- that: ActivityModifyHabit
- myCalendar: Calendar
- requestCode: int
- habitList: ItemList<HiabitType>
- uid: String

onCreate(Bundle): void
+ onDateSet(DatePicker, int, int, int): void
+ onClick(View): void
+ onStart: void
+ onCreateHabitConfirm(View): void
+ onModifyHabitDelete(View): void

Top-right class (yellow):
- mLastKnownLocation: Location
- mLocation: Location
- KEY_CAMERA_POSITION: String
- KEY_LOCATION: String
- M_MAX_ENTRIES: int
- mLikelyPlaceName: String[]
- mLikelyPlaceAddress: String[]
- mLikelyPlaceAttribution: String[]
- mLikelyPlaceLatLng: LatLng

onCreate(Bundle savedInstanceState): void
+ onMapReady(GoogleMap googleMap): void
+ onStart(): void
- getDeviceLocation(): void
onSaveInstanceState(Bundle): void
- getLocatioinPermission(): void
- onRequestPermissionResult(int, String, int[]): void
- updateLocationUI(): void
- AddHabitEventTomMap(): void
-+ isFiveKM(Location, Location): boolean

pCompatActivity

Comparabe<HabitType>

HabitType
- name: String
- reason: String
- startDate: Date
- repeats: boolean[]
- mostRecentEvent: HabitEvent
- userId: String

+ getRepeat(int): boolean
+ setRepeat(int, boolean): void
+ HabitType(String)
+ getName(): String
+ setName(String): void
+ getReason(): String

abitEvent

...ng
...tion

HabitEvents contains
getters and setter,

```
+ REQUEST_NONE: int
+ REQUEST_CREATE_HABIT: int
+ REQUEST_MODIFY_HABIT: int
+ REQUEST_TAKE_PHOTO: int
+ REQUEST_COMPLETE_EVENT: int
+ REQUEST_PICK_IMAGE: int
+ REQUEST_MODIFY_EVENT: int
+ QUERY_CREATE: int
+ QUERY_UPDATE: int
+ QUERY_DELETE: int
+ TASK_SUCCESS: int
+ TASK_FAIL: int
+ TASK_EXCEPTION: int
+ MAX_IMAGE_SIZE: int
+ POLLING_INTERVAL: long

+ FILE_NAME: String
+ ELASTIC_SEARCH_SERVER: String
+ ELASTIC_SEARCH_INDEX: String
+ TYPE_HABIT_TYPE: String
+ TYPE_HABIT_EVENT: String
+ TYPE_USER: String
+ TYPE_FOLLOWINGS: String
+ TYPE_REQUESTS: String
+ EXTRA_ID: String:
+ EXTRA_INDEX: String
+ TIME_FORMAT: SimpleDateFormat
```

```
+ update(int, Object): void
+ process(): void
```

```
- json: String
```

```
ItemHlder(Item)
+ getItem(): Item
```

Application

ConcurrentLinkedQueue

App

- instance: App

+ getContext(): Context
+ onCreate(): void

+ getUid: String
+ generateUid: String
+ getType: String

+ getEncodedImage(): String
+ setEncodedImage(): void

- preEvent: String
- nextEvent: String

+ HabitEvent(String)
+ setEncodedImage(e
+ getComment(): Strin
+ setComment(String)
+ getName(): String
+ getEncodedImage()
+ getHabitLocation():
+ setAttachedLocation
+ isAttachedLocation(
+ setLocation(Integer)
+ compareTo(HabitEv
+ getDate(): Date
+ getStringDate(): Stri
+ getPreEvent(): Strin
+ setPreEvent(String)
+ getNextEvent(): Stri
+ setNextEvent(String
+ getHabitType(): Hab
+ getHabitTypeID(): S
+ setDateForTestingC

IDList

- payload: ArraryList<String>
- isChanged: boolean

+ IDList(String,String)
+ getPayload(): ArrayList<String>
+ isChanged(): boolean
+ setChanged(boolean): void

User

- habitList: ArrayList<String>
- encodeImage: String
- date: String

+ User(String)
+ getEncodeImage(): String
+ setEncodeImage(String): void
+ newHabit(String): void
+ deleteHabit(String): void
+ getDate(): String
+ commit(): void
+ getHabitList(): ArrayList<String>

private name,
startTime, endTime,
comment, date,
photo, ID and
finished.
there is one more
thing: deleteEvent.

encodedImage): void
ng
): void

: String
Location
(boolean): void
): Boolean
: void
vent): int

ing
g
: void
ng
): void
itType
tring
Only(Date): void

+ setReason(String): void
+ getStartDate(): Date
+ setStartDate(Date): void
+ hasEventToday(): boolean
+ getNextEventDay(): Date
+ compareTo(Object): int
+ alreadyDone(): boolean
- isSameDay(Date, Date): boolean
+ getMostRecentEvent(): HabitEvent
+ setMostRecentEvent(HabitEvent): void
+ getUserId(): String