

## ARTIGO: 11382

### Habilitando os Logs

#### Descrição

Nesse artigo veremos como habilitar os logs do Veloster Framework.

#### Habilitando

A api do Veloster disponibiliza um logger, que pode ser modificado de acordo com as necessidades do projeto. Para personalizar o logger, a interface *AppLogger* deve ser implementada:

##### Java Code

```
import java.util.logging.Level;

public interface AppLogger {

    void log(Level level, Class clazz, String message);

    void log(Level level, Class clazz, String message, Exception e);

    void log(Level level, Class clazz, Exception e);
}
```

Então você deve registrar seu logger:

##### Java Code

```
MMLogger.addLogger(new MyLogger());
```

Ou então, você pode usar a implementação padrão disponibilizada na biblioteca *mobilemind-droid-util*, que já é usada pelo *veloster-droid*, da seguinte forma:

##### Java Code

```
MMLogger.addLogger(AppLogger.getInstance());
```

Nesse trecho de código estamos registrando um logger padrão para uso do Veloster. Então já podemos usar o logger:

##### Java Code

```
AppLogger.error(getClass(), "mensagem de erro");
ou
AppLogger.error(getClass(), ex);
```

O logger padrão disponibilizado é implementado com a api Log4j. Para configurar o logger, deve ser criado um arquivo de propriedades chamado *log4j.properties* nos recursos do projeto com o seguinte conteúdo:

##### Java Code

```
log.use.external.storage=TRUE
log.file.name=mobilemind.log
log.file.path.name=mobilemind
log.debug.mode=FALSE
log.max.size.file=2000000
```

Esses são os valores padrão assumidos caso o arquivo não seja criado.