# Aplicació dels terminals

La aplicació està feta en Flutter. Es un entorn de programació que permet construir aplicacions que funcionen en Android, iOS, MacOS I Web sen haver de modificar masses coses (un sol font amb alguns if però coses molt específiques).

Flutter es programa en un llenguatge força modern que es diu Dart I disposa de un bon suport de procediments asíncrons, cosa imprescindible en aplicacions de Android e iOS.

La aplicació està a [GitHub](https://github.com/fgorina/simposi23) per el control de versions I modificacions. Al directori Doc es troba la documentació I el codi per el servidor (simposi23.php) I el script per construir la base de dades (database.mysql).

La aplicació s’estructura a partir de una sèrie de classes bàsiques :

* Server, HashException: Gestiona la comunicació amb el servidor
* Database/Table/DatabaseRecord: Manté les dades en local I memòria, permet actualitzar-les modificar-les I llegir-les I gestiona quan hi han interrupcions amb el servidor. També inclou el codi per les operacions com registrar, comprar I consumir. Es el model de les dades que tenim.
* Participant, Contractació, Producte, Servei, Compra son subclasses de DatabaseRecord que fem servir .
* Utilitats diverses com
  + Scanner : gestiona l’Escàner
  + IconAndFilesUtilities (imatges)
  + LabeledSegments/Switch/TextField (widgets de pantalla)
  + SlideRoutes (moviment de pantalles)
* Pantalles (Views sobre les dades)
  + ParticipantListWidget : Llista de Participants. Gestiona la recerca, escàner i es fa servir per el registre, cerca de participants i consum i exportació.
  + ParticipantView : Una view de un sol participant
  + RespostaWidget : View de resposta de consumir
  + ComprarWidget : View de compra (detalls del producte, preu etc)
  + CompresListWidget : Llista de compres (i exportació)
  + SeveiListWidget : Llista de Serveis
  + BacklogListWidget : Consulta i gestió del Backlog (operacions pendents de enviar al servidor).

**Comunicació entre objectes**

Hi ha dos camins principals:

* Quan obrim una view passem dades com la id etc.
* Quan es reben dades s’actualitza la base de dades, es salva una còpia en local I s’envia una notificació a totes les views subscrites que poden, si cal, actualitzar-se. Això pot succeir quan tenim unes dades presents a pantalla I s’acaba de executar un request al servidor que modifica les dades.

**Limitacions**

No tenim cap mecanisme que permeti fer un push des de el servidor quan s’actualitza alguna dada per part de un altre terminal, ara bé si que s’executa un request general cada vegada que l’aplicació retorna de estar en segon pla (per exemple, quan la pantalla s’ha bloquejat per massa temps d’inactivitat, hem canviat d’aplicació I tornem etc.)

Tot això es irrellevant quan tenim connectivitat doncs :

* Al obrir les dades de un participant fem un request de les dades al servidor
* Al consumir un servei demanem al servidor si tot es ok

Però pot afectar si no tenim connectivitat I tenim varis terminals operant simultàniament. Ex si tenim dos entrades per esmorzar I el participant passa una vegada per entrada 1 I un altre per entrada 2. Si ha connectivitat ho detecta però si no es pot escapar. Es un cas una mica rebuscat oi?