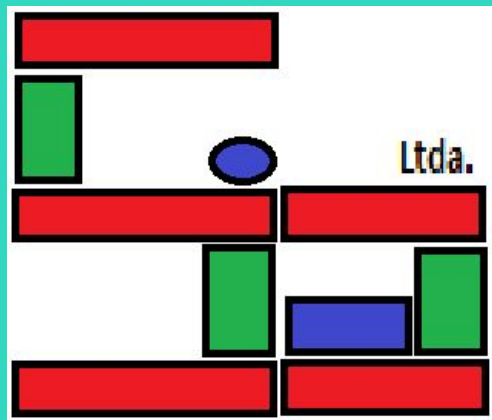
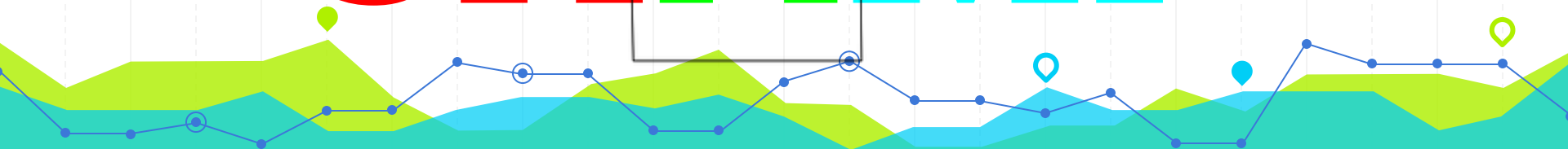


SHAMP



EQUIPO DE TRABAJO



Nicolas Ordoñez
Lider de arquitectura



David Gomez
Lider de desarrollo de
activos



Sergio Bravo
Lider de desarrollo de
productos

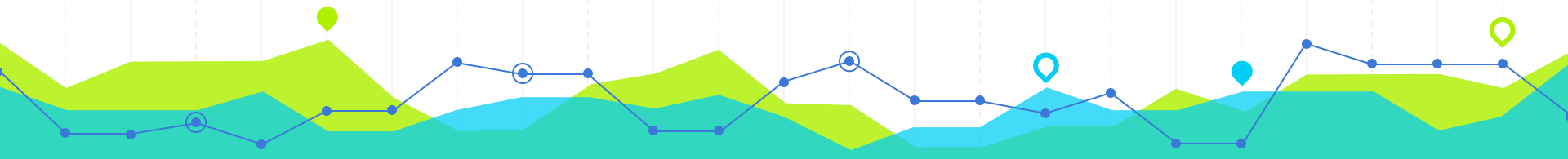


Fredy Novoa
Líder del proyecto



Contenido

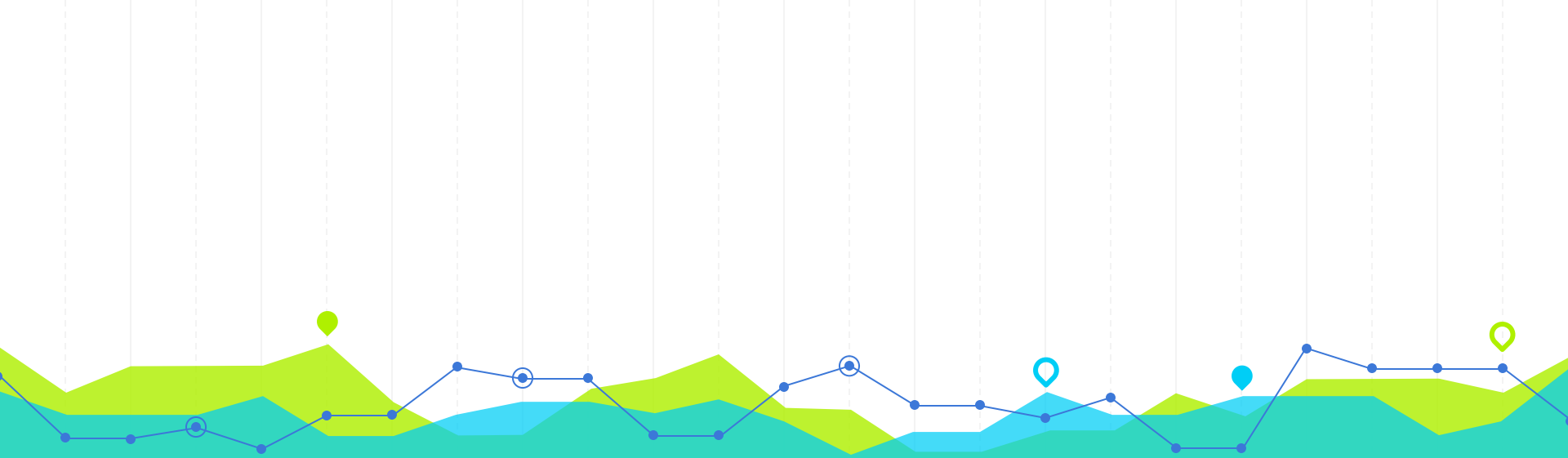
1. **Una vista atrás**
 - a. **Nuestro producto final**
 - b. **Propuestas de valor**
 - c. **Vista de capas**
2. **Derivador**
 - a. **Estructura**
 - b. **¿Qué se tuvo en cuenta?**
3. **Formas de implementar la derivación**
4. **Presentación página web + móvil**
5. **Estrella de mar**
 - a. **Comparación antes VS resultados**



Una vista atrás

¿Qué pasó antes en SHAMP?

1



Nuestro producto final

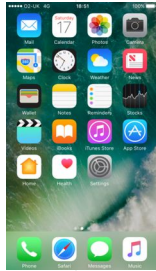
¿Cómo se usa SHAMP?



SHAMP (producto entregable)

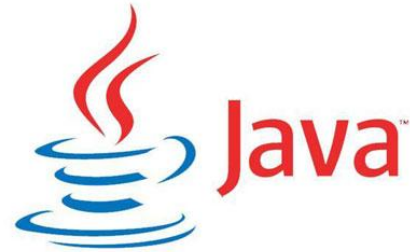
Aplicación

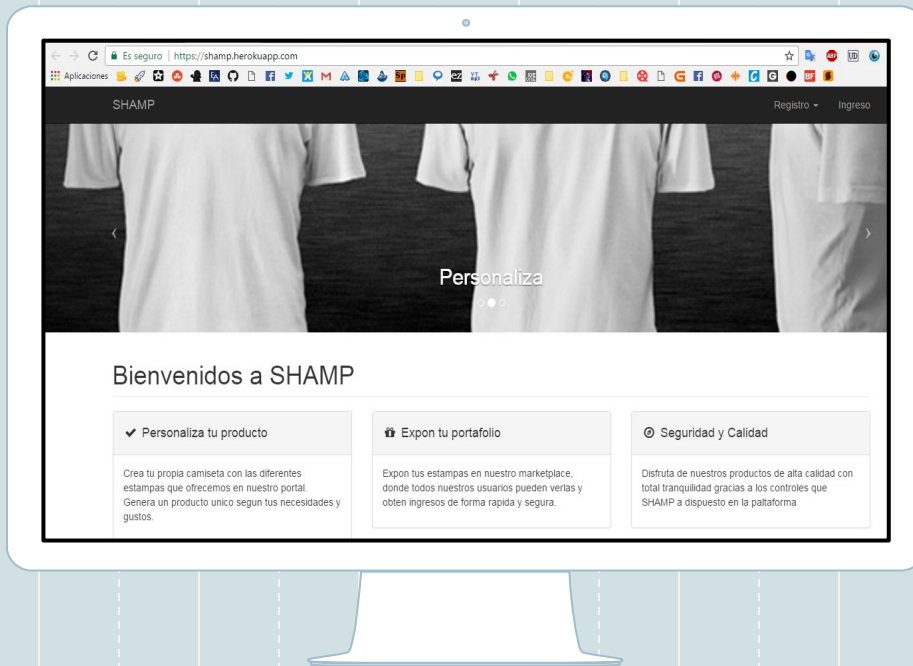
- ◉ Permite compras simuladas.
- ◉ Diseño de estampados en camisas.
- ◉ iOS App.



Back end

- Realización de operaciones



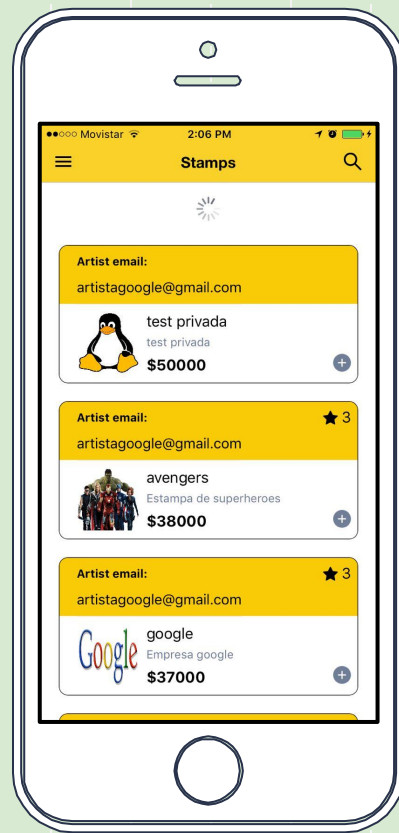


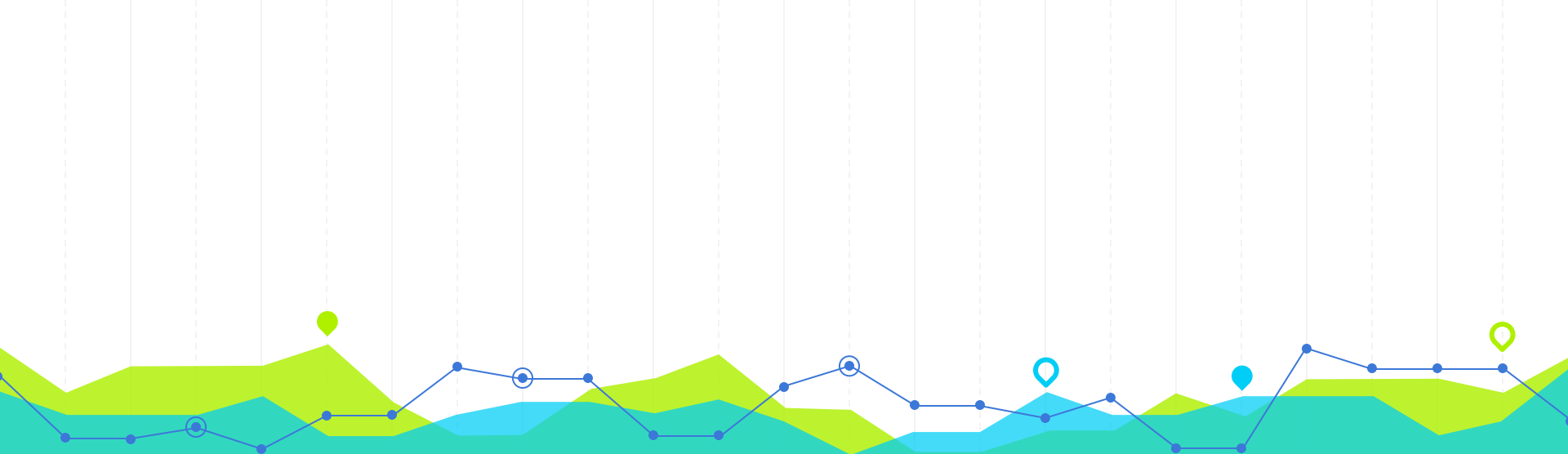
Desktop

Consola de administración está presente en la web

iPHONE

Aplicación corre en iOS para el cliente





Nuestra propuesta de valor

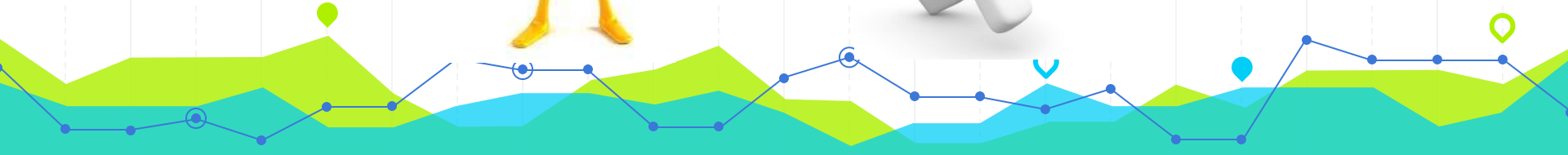
¿Qué ofrece SHAMP diferente?



SHAMP (canvas)

Propuestas de valor

- ◉ Estampado Personalizado.
 - ◉ Construcción de diseños personalizados.

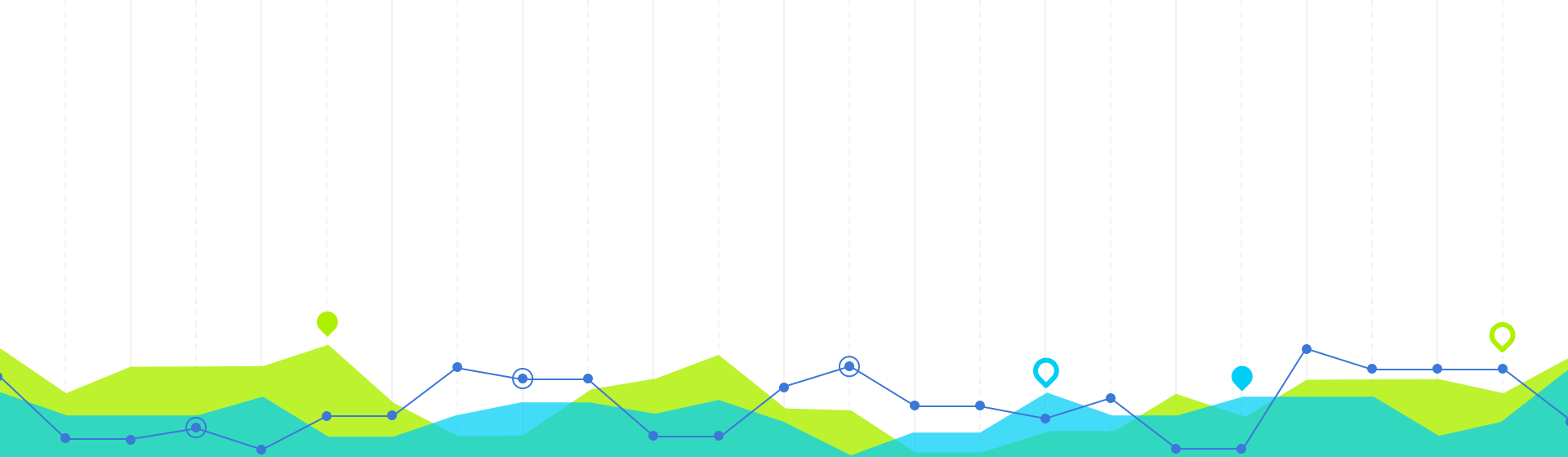


SHAMP (canvas)

Propuestas de valor

- Imagen con filtros.
 - Seleccionar filtros
 - Brindar mayor diversidad al momento de escoger un diseño





Arquitectura

La vista de capas muestra cómo fue el diseño
del proyecto

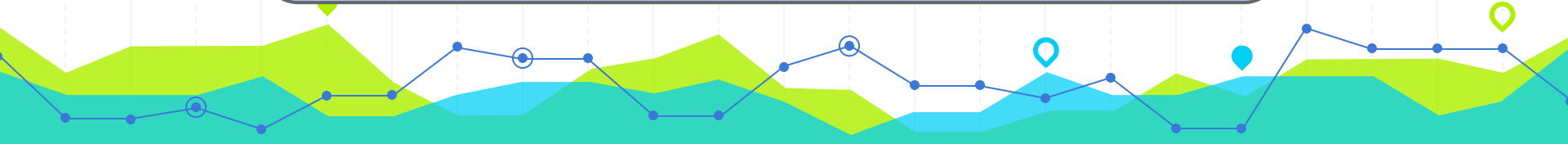


iOS

Web

Lógica del negocio

Base de datos



iOS

Web

Swift

Apk instalada en
dispositivo se conecta
por internet



iOS

Web

HTML y CSS
Play framework en java
Desplegado en Heroku



Play framework en java

Lógica del negocio

Servidor de aplicaciones
en Heroku





MySQL
Desplegado en
Amazon

Base de datos

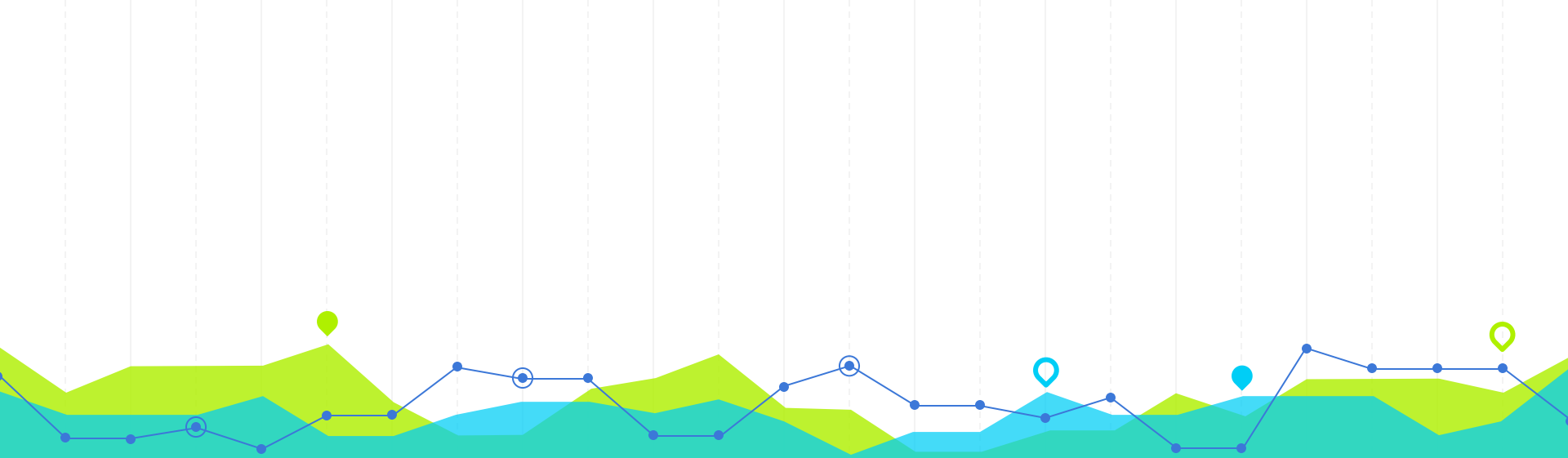


1

Derivador

¿Cómo se desarrollo?

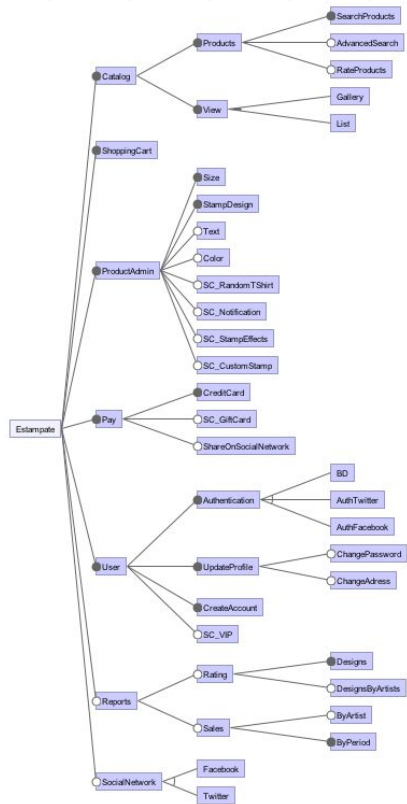




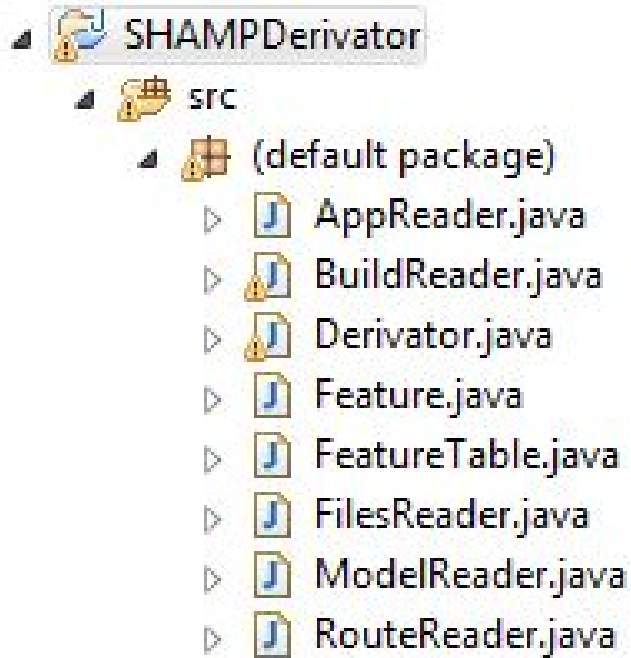
Estructura

¿Cómo está organizado?





Modelo de
variabilidad
desarrollado en
Feature IDE cómo
base



Separación de las
clases según el
tipo de
modificación que
se necesitaba
hacer

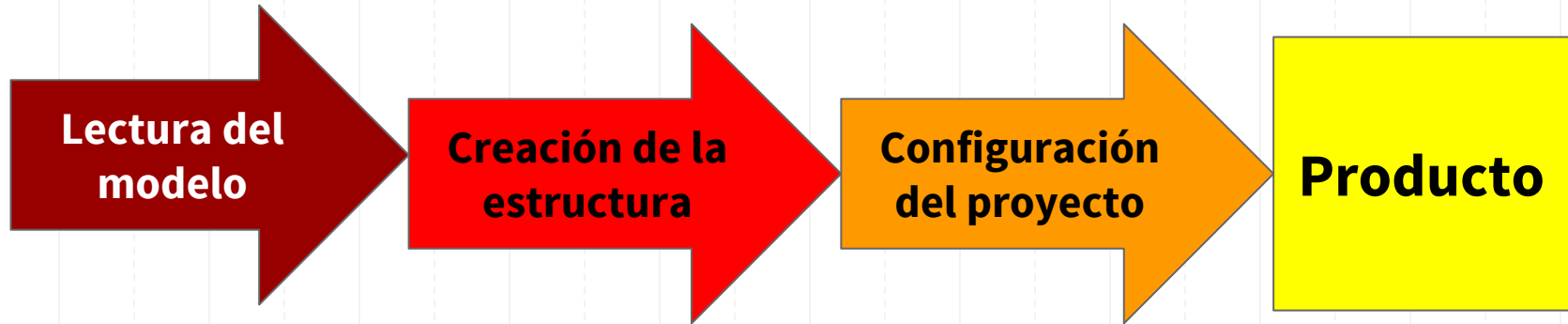
Se representó el feature para organizar la variación de manera más sencilla.

```
public class Feature {  
  
    String name = "";  
    boolean state = false;  
  
    public Feature(String Name, boolean State) {  
        name = Name;  
        state = State;  
    }  
}
```

```
public class FeatureTable {  
  
    private Hashtable<String, Boolean> optionals;  
  
    public FeatureTable() {  
        optionals = new Hashtable<>();  
    }  
  
    public Hashtable<String, Boolean> getTable() {  
        return optionals;  
    }  
  
    public void addFeature(String name, boolean state) {  
        optionals.put(name, state);  
    }  
}
```



Se representó el feature para organizar la variación de manera más sencilla.



```
public class ModelReader {  
  
    public static FeatureTable readModel(String file, FeatureTable optional)  
        throws IOException, NoSuchAlgorithmException {  
        BufferedReader br = new BufferedReader(new FileReader(file));  
        String begin = "";  
        begin = Derivator.FEATURE;  
    }  
}
```



¿Qué se tuvo en cuenta?

¿Cómo se desarrollo?

b

build.sbt

```
/**
 * Lectura y escritura del archivo build.sbt
 *
 * @param file
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
public static void readBuild(String file, FeatureTable optional)
    throws IOException, NoSuchAlgorithmException {
    BufferedReader br = new BufferedReader(new FileReader(file));
    StringBuffer sb = new StringBuffer();
}
```

¿Que lugares se debian modificar?



Routes

```
/**
 * Evalua unos parametros para cambiar el archivo de rutas
 *
 * @param nameFeature
 * @param lineEnter
 * @param start
 * @param end
 * @return
 */
private static String evaluateRoute(String nameFeature, String lineEnter,
String feature, FeatureTable optional, String file) {
    String lineExit = lineEnter;
    if (!optional.getOptional(nameFeature).getState()) {
        if (lineExit.equals(Derivator.END_ROUTE + " " + feature)) {
            Derivator.flagRoute = false;
            Derivator.actualFeature = "";
        }
    }
}
```

Views

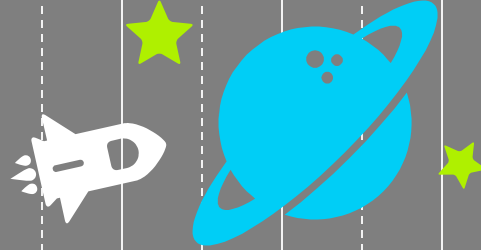
```
/**
 * Incluye o excluye vistas dependiendo del valor de un feature
 *
 * @param file
 * @param feature
 * @param optional
 */
public static void exincludeViews(String file, String feature,
FeatureTable optional) {
    String directionFile = file.replace("/", "\\").replace("build.sbt", "");
    String directionFileTo = directionFile + "app\\views\\";
    if (!optional.getOptional(feature).getState()) {
        switch (feature) {
            case "Notification":
                findExcludeFiles(directionFileTo, Derivator.MENSAJE_VIEWS);
        }
    }
}
```

¿Que debía estar y que no?

Application .conf

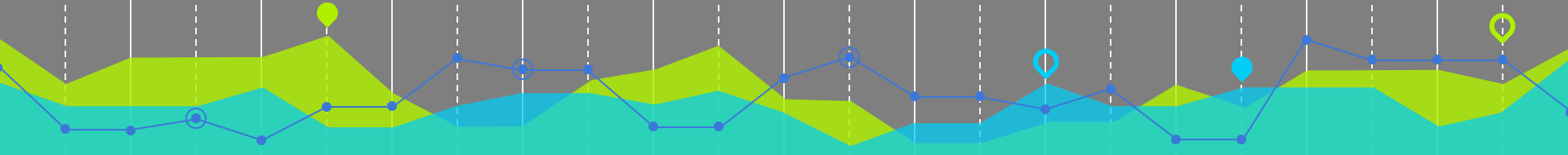
```
/**
 * Lee el application conf del proyecto, y cambia las líneas según la
 * configuración realizada en feature IDE
 *
 * @param file
 * @throws IOException
 * @throws NoSuchAlgorithmException
 *
 *      Ejemplo: line = evaluateRoute("Notification", line, MESSAGE,
 *      actualFeature);
 */
public static void readApplicationConf(String file, FeatureTable optional)
    throws IOException, NoSuchAlgorithmException {
    BufferedReader br = new BufferedReader(new FileReader(file));
    StringBuffer sb = new StringBuffer();
    String begin = "";
    begin = Derivator.FEATURE_CONF;
```

¿Donde se iban a realizar los cambios?



Derivación

¿En que forma se implemento?



Atributos trabajados en el derivador

1. Mensajes con dependencias
2. Filtros con patrón decorador
3. Texto y color con patrón Builder
4. Cambio de password con constante sobre variable



Constante sobre variable

1. Búsqueda avanzada
2. Cambio de password
3. Login Redes sociales
4. Compartir Redes sociales
5. Notificaciones



Patrones de software empleados

1. Patron decorador

Se utilizó para realizar la derivación de los filtros por cada estampa.

2. Patron Builder

Se utilizó para realizar la derivación de texto en una orden de compra.



Variabilidad en servicios REST

Se utilizó derivación sobre los REST en las características de:

1. Cambio de password
2. Estampa privada
3. Notificación
4. Reportes

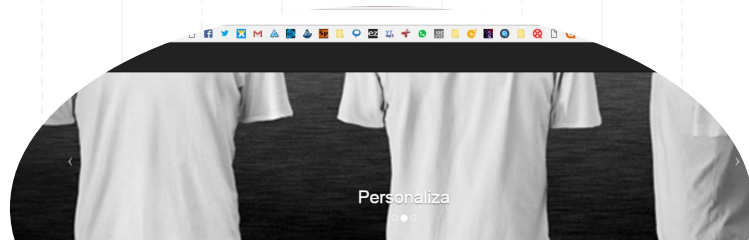




3

Presentación en vivo

SHAMP



Bienvenidos a SHAMP

✓ Personaliza tu producto

... camiseta con las diferentes
... mos en nuestro portal.
... tus necesidades y

🏠 Expon tu portafolio

Expon tus estampas en nuestro marketplace,
donde todos nuestros usuarios pueden verlas y
obten ingresos de forma rapida y segura.

🔒 Seguridad y Calidad

Disfruta de nuestros pro-
total tranquilidad
SHAMP a

Desplegado en Heroku

<https://shamp.herokuapp.com/>



WORK HARDER

4

Estrella de mar

Ítem

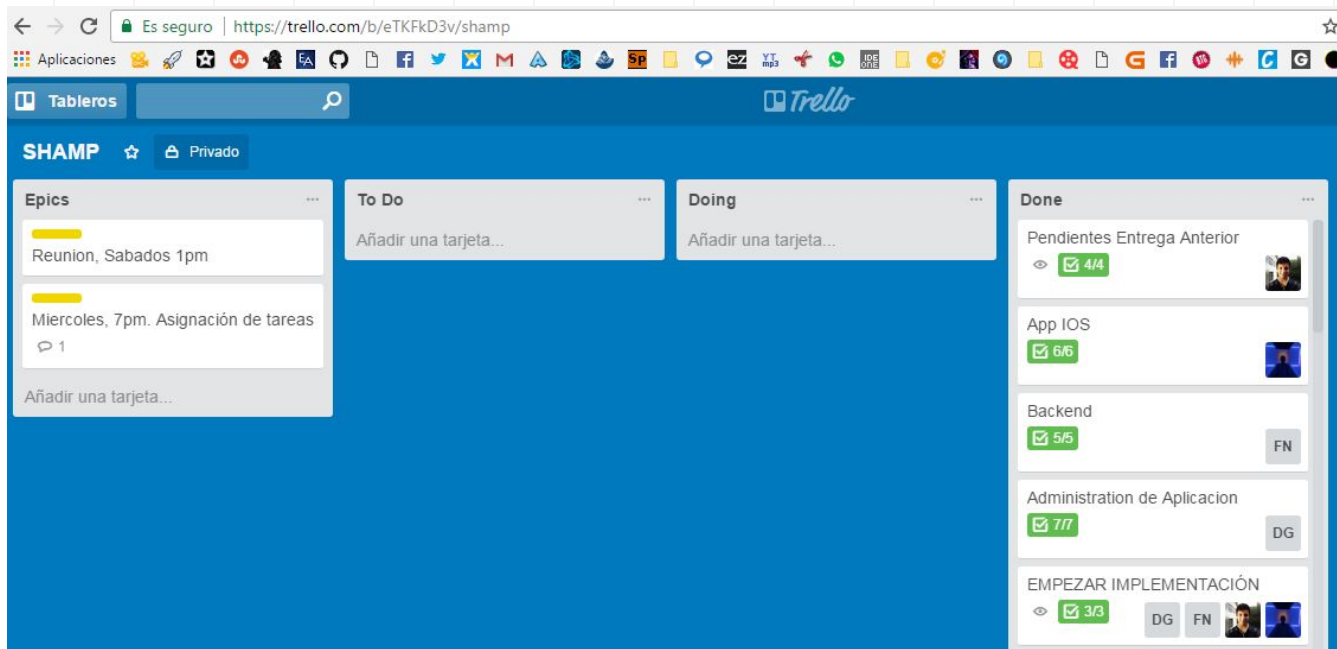
Planeación de actividades

Compromisos con las fechas y tareas asignadas

Utilización de metodologías ágiles

Mayor transferencia de conocimiento

Mas de ...



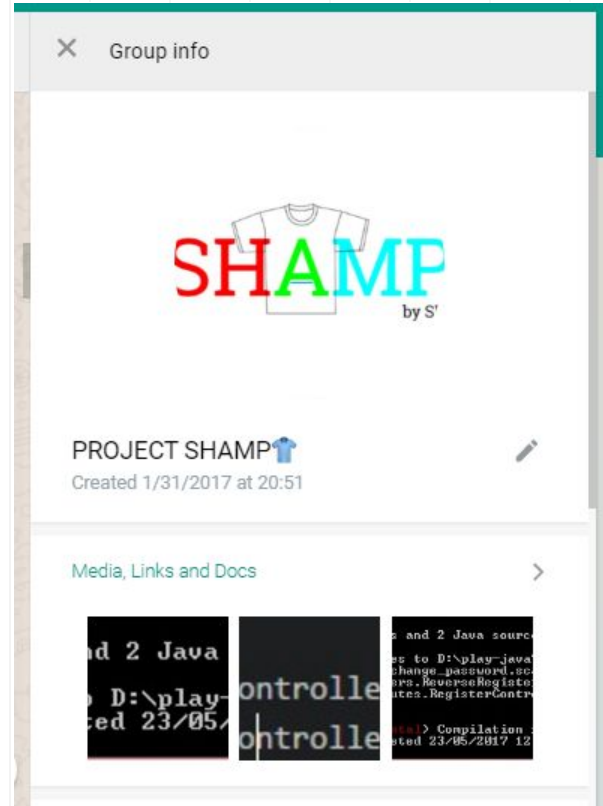
Seguir Haciendo ...

Ítem

Mantener comunicación fluida

Mantener el esquema de infraestructura de pruebas

Continuar utilizando herramientas de colaboración



Dejar de hacer ...

```
a-2.11\classes...
[info] Compiling 35 Scala sources to D:\play-java\target\scala-2.11\classes...
[success] Total time: 103 s, completed 23/05/2017 01:02:12 PM
[play-java] $ run
[info] Setting up Play fork run ... (use Ctrl+D to cancel)
[info] Updating {file:/D:/play-java/myProject...
[info] Resolving com.fasterxml.jackson.datatype#jackson-datatype-jsr310;2.7.1 ..
[info] Resolving com.fasterxml.jackson.datatype#jackson-datatype-jsr310;2.7.8 ..
[info] Resolving com.fasterxml.jackson.dataformat#jackson-dataformat-cbor;2.6.6 ..
[info] Resolving com.amazonaws#aws-java-sdk-marketplacecommerceanalytics;1.11.111 ..
[info] Resolving com.amazonaws#aws-java-sdk-marketplacemeteringservice;1.11.111 ..
[info] Resolving com.amazonaws#aws-java-sdk-mechanicalturkrequester;1.11.111 ..
[info] Resolving org.eclipse.jetty.websocket#websocket-client;9.2.15.v20160210 ..
[info] Resolving org.eclipse.jetty.websocket#websocket-common;9.2.15.v20160210 ..
[info] Resolving net.contentobjects.jnotify#jnotify;0.94-play-1 ...
[info] Done updating.
[warn] There may be incompatibilities among your library dependencies.
[warn] Here are some of the libraries that were evicted:
[warn] * com.typesafe.play:play-json_2.11:(2.5.2, 2.3.4) -> 2.5.10
[warn] Run 'evicted' to see detailed eviction warnings
[info]
--- (Running the application, auto-reloading is enabled) ---
[info]
[info] p.c.s.NettyServer - Listening for HTTP on /0:0:0:0:0:0:0:0:9000
[info]
```

Ítem

Desarrollos
independientes

Toma de decisiones
unilaterales

Utilización de
repositorios locales

Comenzar a hacer ...

Ítem

Pruebas Integrales

Construir agenda de las reuniones

Solicitar información requerida

Documentar código

```
//Constantes con las vistas que dependen de un feature
public static final String[] MENSAJE_VIEWS = {"adminMessages.scala.html", "res
public static final String[] REPORTS_VIEWS = {"create_report.scala.html", "vie
public static final String[] CHANGEPASSWORD_VIEWS = {"change_password.scala.ht
/**
 * Metodo principal del derivador que ejecuta los metodos en un orden
 * determinado para cambiar algunos aspectos en la aplicaiación
 *
 * @param args
 */
public static void main(String[] args) {
    // TODO Auto-generated method stub
    try {

        // Se lee el modelo creado en FeatureIDE
        System.err.println("Inicia la lectura del modelo de FeatureIDE...");
        optional = ModelReader.readModel(
```

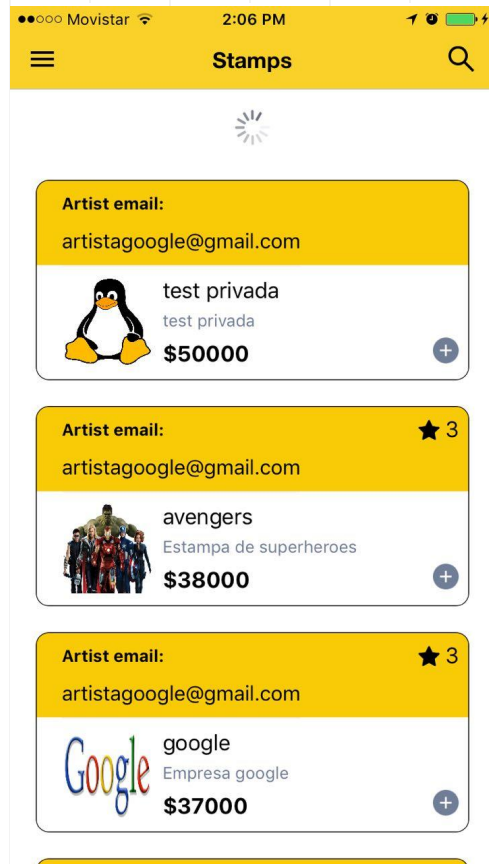

Menos de ...

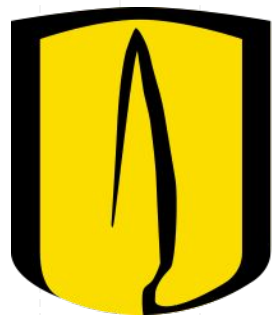
Ítem

Trabajo sobre la fecha de entrega

Subestimar los errores de presentación

Trabajo sin tener en cuenta diseño





Universidad de los Andes

¡Gracias!



¿Preguntas?

Más información en <https://github.com/SPL-201710/SHAMP/wiki/>



Referencias



- Información de la aplicación en la Wiki de Shamp
<https://github.com/SPL-201710/SHAMP/wiki/>
- Presentation en SlidesCarnival
- Imagenes by Unsplash

