

Resumo

Os Smartphones e os Tablets são dispositivos em expansão e o número de utilizadores aumenta a cada dia que passa. A capacidade crescente de processamento, armazenamento e de incorporação de tecnologias de comunicação nestes dispositivos, representa um potencial enorme para as empresas que trabalham neste mercado, especialmente as que se dedicam ao desenvolvimento de aplicações móveis. Paralelamente, a evolução dos canais de comunicação permite a transferência de dados a débitos crescentemente superiores e de uma forma cada vez mais transparente e independente da tecnologia de acesso.

Este relatório insere-se no âmbito de um estágio curricular realizado na área das telecomunicações e multimédia, descrevendo o acompanhamento do desenvolvimento de uma aplicação para duas plataformas: iPhone (Smartphone) e iPad (Tablet). A aplicação tem como objectivo disponibilizar várias formas de comunicação (chat, transferência de ficheiros, chamadas de voz, entre outras) de uma forma interoperável com outras aplicações que obedeçam à mesma especificação protocolar RCSe (Rich Communications System enhanced), uma iniciativa do grupo GSMA.

O primeiro semestre consistiu numa integração do estagiário na equipa responsável pela aplicação para iPhone. No segundo semestre foram implementadas algumas funcionalidades adicionais nessa mesma aplicação, e foi iniciado o desenvolvimento de raiz da aplicação para iPad.

Palavras-Chave

“Aplicação”, “Interoperabilidade”, “Smartphone”, “Tablet”, “iPhone”, “iPad”, “iOS”, “RCSe”, “GSMA”, “SIP”, “MSRP”

Índice

Índice de imagens	1
Lista de Acrónimos.....	3
Capítulo 1 Introdução	5
Capítulo 2 Estado da Arte	7
2.1. Aplicações Over-The-Top	7
Serviços inovadores.....	7
Aplicações mais vendidas.....	9
Comparativo das aplicações.....	12
2.2. A resposta da GSMA	13
Reconhecimento das forças e fraquezas.....	13
Restituição da proposta de valor: o projecto RCE	14
2.3 A especificação RCS-e / Joyn.....	14
Funcionalidades	14
A arquitectura IMS.....	15
O protocolo SIP	16
O protocolo MSRP	17
Lançamentos	18
Capítulo 3 Objectivos do Estágio e Método de Abordagem	19
3.1 A WIT Communications Suite.....	19
3.2 A aplicação WMC.....	20
Arquitectura dos clientes.....	21
3.3 Objectivos para o WMC iPhone	22
3.4 Objectivos para o WMC iPad.....	23
Funcionalidades inicialmente previstas	23
Realocação para o WMC iPhone RCS2.....	23
Regresso ao WMC iPad.....	24
3.5 Metodologia de desenvolvimento	24
Metodologia Waterfall	24
Metodologia Agile	25
A metodologia escolhida	28
3.6 Ferramentas utilizadas.....	28
Capítulo 4 Plano de Trabalho e Implicações.....	31

Capítulo 5 Resultados.....	37
5.1 WMC iPhone – 1º Semestre	37
Image Share numa chamada nativa	37
Cancelamento do Sign In.....	40
Edição do avatar de um contacto	41
Histórico de comunicação de um contacto	42
5.2 WMC iPhone – 2º Semestre	43
Image Share numa chamada VoIP	43
Video Share numa chamada VoIP	44
5.3 WMC iPad.....	45
Arquitectura da aplicação	45
Esqueleto da interface com o utilizador	47
Módulo de sessão	47
Módulo de capabilities	50
Módulo de chat	50
Capítulo 6 Conclusões.....	52
Referências	53

Índice de imagens

Figura 1 - Comparação das aplicações OTT por funcionalidades que disponibilizam	12
Figura 2 - Comparação das aplicações OTT por plataformas para as quais estão disponíveis	13
Figura 3 - Pilha de serviços propostos pela especificação RCS-e.....	15
Figura 4 - Arquitectura IMS (IP Multimedia Subsystem)	16
Figura 5 - Exemplo da sequência de troca de mensagens SIP no registo de um utilizador..	17
Figura 6 - Compromisso de diferentes países em lançar o RSC-e em 2012.....	18
Figura 7 - Arquitectura global da WIT Communications Suite	19
Figura 8 - Diagrama de arquitectura dos clientes WMC iPad e WMC iPhone.....	21
Figura 9 - Diagrama de fluxo da Metodologia Waterfall.....	24
Figura 10 - Diagrama de fluxo da Metodologia Agile	26
Figura 11 - IDE Xcode, com o Interface Builder aberto.....	29
Figura 12 - Wireshark numa captura utilizando um filtro “sip”	29
Figura 13 - Visualização de um repositório no svnX	30
Figura 14 - Planeamento inicial para o 1º semestre	32
Figura 15 - Planeamento inicial para o 2º semestre	33
Figura 16 - Planeamento actual para o 1º semestre	34
Figura 17 - Planeamento em Janeiro para o 2º semestre	35
Figura 18 - Planeamento final para o 2º semestre.....	36
Figura 19 - Screenshots do image share na chamada nativa no WMC iPhone 1/3.....	37
Figura 20 - Screenshots do image share na chamada nativa no WMC iPhone 2/3.....	38
Figura 21 - Screenshots do image share na chamada nativa no WMC iPhone 3/3.....	39
Figura 22 - Screenshots do cancelamento do sign in no WMC iPhone	40
Figura 23 - Screenshots da edição do avatar de um contacto no WMC iPhone.....	41
Figura 24 - Screenshots do histórico de comunicação de um contacto no WMC iPhone	42
Figura 25 - Screenshots do Image Share em VoIP no WMC iPhone.....	43
Figura 26 - Screenshots do Video Share em VoIP no WMC iPhone	44
Figura 27 - Diagrama de classes da aplicação WMC iPad	46
Figura 28 - Esqueleto da UI do WMC iPad.....	47

Figura 29 - Perda de ligação na aplicação WMC iPad	48
Figura 30 - Definições da aplicação nas Definições Gerais do iPad	48
Figura 31 - Adição de um contacto novo no WMC iPad	49
Figura 32 - Edição de um contacto no WMC iPad	50
Figura 33 - Vista de chat no WMC iPad.....	51

Lista de Acrónimos

2G	2 nd Generation of GSM
3G	3 rd Generation of GSM
4G	4 th Generation of GSM
3GPP	3 rd Generation Partnership Project
ACK	Acknowledgement
API	Application Programming Interface
AS	Application Server
CPIM	Common Profile for Instant Messaging
CS	Circuit Switched
CSCF	Call Session Control Function
DSL	Digital Subscriber Line
EAB	Enhanced Address Book
ESB	Enterprise Service Bus
GPRS	General Packet Radio Service
GSM	Global System for Mobile communication
GSMA	Groupe Speciale Mobile Association
GW	Gateway
HD	High Definition (voz ou vídeo)
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
IETF	Internet Engineering Task Force
IDC	International Data Corporation
IDE	Integrated Development Environment
IM	Instant Messaging
IMDN	Instant Message Disposition Notification
IMS	IP Multimedia Subsystem
IP	Internet Protocol
IS	Image Share
ISP	Internet Service Provider
LTE	Long Term Evolution (4G)
MCC	Mobile Country Code
MNC	Mobile Network Code
MMS	Multimedia Messaging Service
MNO	Mobile Network Operator
MPEG	Moving Pictures Experts Group
MSISDN	Mobile Subscriber Integrated Services Digital Network Number
MSRP	Message Session Relay Protocol
NAT	Network Address Translation
NW	Network
OMA	Open Mobile Alliance
OSA	Open Services Architecture Gateway
OTT	Over-The-Top
OS	Operating System
PDA	Personal Digital Assistant
PS	Packet Switched
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RCE	Rich Communications Ecosystem

RCS	Rich Communications Suite
RCS_e	Rich Communications Suite enhanced
RFP	Request For Proposal
RTP	Real-time Transport Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SIM	Subscriber Identity Module
SMS	Short Message Service
TCP	Transmission Control Protocol
TEL URI	TELEphone URI
UDP	User Datagram Protocol
UE	User Equipment
UI	User Interface
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UUID	Universally Unique Identifier
UX	User Experience
VoIP	Voice over IP
VoLTE	Voice over LTE
VS	Video Share
XCAP	XML Configuration Access Protocol
XDM	XML Document Management
XDMS	XML Data Management Server
XML	Extensible Markup Language
Wi-Fi	Synonym for WLAN
WLAN	Wireless Local Area Network
WCAS	WIT Communication Application Server
WCS	WIT Communications Suite
WMC	WIT Mobile Communicator
WPC	WIT PC Communicator
WWC	WIT Web Communicator

Capítulo 1

Introdução

Os progressos tecnológicos têm vindo a reinventar o estilo de vida do ser humano, e um dos exemplos que melhor constata esta realidade é a evolução dos dispositivos móveis.

Os Smartphones e os Tablets infiltraram-se no quotidiano das pessoas. De acordo com a IDC (International Data Corporation), em Junho de 2011 previa-se que no final do ano tivesse sido vendido em todo o mundo um total anual de 472 milhões de smartphones, e espera-se que no ano de 2015 esse número venha a aumentar para perto de um bilião [1]. Em simultâneo, outro estudo observa um enfraquecimento do mercado de Notebooks em paralelo com um crescimento do mercado de Tablets, sugerindo uma migração de um dispositivo para o outro [2].

Hoje estes dispositivos representam uma ferramenta para um variadíssimo e crescente leque de actividades, acompanhando os utilizadores para onde quer que se desloquem. No entanto, de todas estas actividades existe uma que se destaca como a mais antiga de todas, assumindo um papel de tal forma importante que se encontra presente em todos os telemóveis e tablets produzidos até hoje: a comunicação.

Também a comunicação tem vindo a reinventar-se de acordo com os progressos tecnológicos. Do tradicional telefone fixo ao smartphone passámos de uma chamada de voz entre pontos geograficamente restrictos a um conjunto vasto de formas de comunicação com uma abrangência cada vez mais próxima de uma abrangência global. Hoje podemos realizar chamadas, trocar mensagens, partilhar conteúdos audio-visuais e muito mais, praticamente em qualquer local do mundo. Este enriquecimento da comunicação tem vindo a moldar os comportamentos das pessoas, uma realidade que se manifesta actualmente na ansiedade de utilização das redes sociais e do instant messaging [3].

Esta diversidade nas formas de comunicação foi pioneiramente disponibilizada pelos ISP¹s por redes de cabo (coaxial e fibra óptica), mas as limitações destas redes impossibilitam a satisfação de uma das maiores exigências dos utilizadores: a mobilidade. Conscientes destas exigências, os operadores das redes celulares passaram a disponibilizar serviços de dados pela rede 3G, o que abriu um mercado para a comercialização de aplicações para dispositivos móveis que dispusessem de acesso a esta rede.

Este mercado tem vindo a ser fortemente explorado pelas aplicações OTT (Over-The-Top): aplicações desenvolvidas por empresas não relacionadas com os operadores móveis e que o utilizador pode instalar no seu Smartphone. Estas aplicações em norma tomam partido dos pacotes de dados que o utilizador subscreve, não sofrendo qualquer influência ou controlo dos operadores. Como muitos dos serviços disponibilizados por estas aplicações são grátis e permitem substituir os serviços disponibilizados pelos operadores, estas empresas constituem uma forte ameaça para os operadores móveis.

Em resposta a esta ameaça, a GSMA [4] (Groupe Speciale Mobile Association) um grupo que representa os interesses dos operadores móveis em todo o mundo desenvolveu um projecto de nome RCE [5] (Rich Communications Ecosystem) com o intuito de enriquecer os serviços de comunicação disponibilizados pelas redes móveis. Este projecto assenta na especificação RCS-e (Rich Communication Suite-enhanced) que pretende estender as comunicações por voz e SMS disponibilizando instant messaging, partilha de vídeo em

¹ Internet Service Provider

tempo real e transferência de ficheiros entre qualquer dispositivo, em qualquer rede, e com qualquer contacto da lista telefónica. O RCS-e é comercialmente conhecido como **Joyn**.

Foi neste contexto que se inseriu o desenvolvimento da aplicação proposto neste estágio. A aplicação consiste num comunicador enriquecido que segue a especificação RCS-e, disponibilizando as funcionalidades acima descritas, entre outras adicionais. Apesar de ainda não ter sido lançada no mercado, a aplicação Joyn para iPhone foi já distinguida pela GSMA como a aplicação mundial de referência Joyn.

O estagiário teve a seu cargo o desenvolvimento de algumas funcionalidades da aplicação para iPhone, e o desenvolvimento de raiz da aplicação para iPad, encontrando-se sempre inserido numa pequena equipa de desenvolvimento (2 a 4 colaboradores).

Este estágio curricular foi proposto pela empresa WIT Software [6] ao Departamento de Engenharia Informática da Universidade de Coimbra, e encontra-se sob a orientação do professor Nuno Pimenta (no DEI) e do gestor do produto Frederico Lopes (na empresa). A realização do estágio decorreu na sede da empresa localizada em Banhos Secos, Coimbra.

Capítulo 2

Estado da Arte

A predominância dos Smartphones no mercado veio aumentar drasticamente o número de downloads de aplicações móveis. Com a tendência crescente que se tem observado, prevê-se [7] que em 2014 este número atinja os 76.9 biliões, e que gere um total de receitas de 35 biliões de dólares americanos. Esta competição feroz obriga muitas empresas a seguirem uma filosofia de “aplicações grátis”, alterando o modelo de negócio para as vendas “in-app” (funcionalidades ou conteúdos extra que o utilizador pode comprar ou aceder dentro da aplicação), para as receitas provenientes de publicidade, ou para as receitas provenientes de subscrições mensais.

Para os operadores móveis este crescimento tem-se revelado uma forte ameaça. A inovação nos serviços de comunicação das aplicações Over-The-Top, aliada à natureza gratuita dos mesmos, tem provocado uma migração dos seus subscritores para este tipo de aplicações. Os operadores continuam a ser os maiores fornecedores das comunicações tradicionais, e um estudo prevê [8] que em 2013, 84% das trocas de mensagens no mundo em dispositivos móveis sejam realizadas por SMS, 10% por email, 5% por MMS e 1% por instant messaging. No entanto, o mesmo estudo revela que o crescimento do SMS está a desacelerar rapidamente em oposição ao crescimento do email e instant messaging nos dispositivos móveis.

Neste capítulo começa-se por fazer um estudo das aplicações Over-The-Top: identificação dos serviços inovadores que mais ameaçam os operadores, caracterização das aplicações mais vendidas que fornecem esses serviços, e estabelecimento de um comparativo entre as mesmas.

Em seguida é explicada a resposta dos operadores móveis a esta ameaça: o projecto Rich Communications Ecosystem (RCE). Será feita uma breve dissecação deste projecto, ilustrando os objectivos, os passos que pretendem seguir e o planeamento do mesmo.

Por fim, serão apresentados alguns competidores directos da Wit Software no desenvolvimento de aplicações RCS-e.

2.1. Aplicações Over-The-Top

Serviços inovadores

Os operadores móveis têm disponibilizado três serviços em alta escala: chamadas de voz, SMSs e MMS. Estes serviços cedo se revelaram insuficientes para os utilizadores, e da competição entre as empresas de aplicações móveis para satisfazer as necessidades dos utilizadores nasceram vários serviços apelativos:

- **Chamadas VoIP** - Chamadas de voz efectuadas sobre a rede de dados celular, permitindo ao utilizador pagar o mesmo por uma chamada de longa distância que pagaria por uma chamada local (o que paga é o consumo de dados).
- **Chamadas VoIP em conferência** - São chamadas VoIP onde mais de 2 contactos conversam em simultâneo.
- **Videochamadas VoIP** – Chamadas VoIP com um contacto com partilha de vídeo em simultâneo e em tempo real.

- **Videochamadas VoIP em conferência** – Videochamadas VoIP com mais de 2 contactos em simultâneo.
- **Instant Messaging** – Conversação por escrito em tempo real com um contacto.
- **Group Messaging** – Conversação por escrito em tempo real com vários contactos em simultâneo (todos dentro da mesma “sala”). Algumas aplicações permitem a constituição de grupos pré-definidos, algo que se tem manifestado como muito apelativo para os utilizadores.
- **Presença** – Informação sobre o estado de um contacto relativamente às interações disponíveis a ser realizadas com ele (disponível, ausente, ocupado...).
- **Transferência de ficheiros** – Envio e recepção de ficheiros. Algumas aplicações permitem apenas o envio e recepção de ficheiros multimédia como imagens e vídeos, e a grande maioria destes ficheiros são resultado de uma captura/gravação feita pelo dispositivo. Está [9] estimado que cerca de 25% das capturas/gravações de hoje são realizadas por smartphones, conquistando muito do espaço ocupado pelas máquinas dedicadas a essas funcionalidades.
- **Lista de contactos** – Algumas aplicações utilizam uma lista de contactos local, outras utilizam uma lista de contactos remota. Muitas permitem sincronizar a lista de contactos da aplicação com a lista de contactos nativa ou de uma rede social.
- **Integração com redes sociais** – Além das aplicações móveis das redes sociais, muitas outras tomam partido das APIs² disponibilizadas pelas redes para integração na aplicação. O utilizador pode assim partilhar conteúdos ou aceder à rede social dentro de outra aplicação.
- **Geolocalização** – Partilha da localização geográfica do utilizador, de um evento ou de um local, geralmente referenciada por coordenadas de latitude e longitude.

Estes são os serviços mais comuns, mas outras aplicações podem incluir:

- Serviços de voicemail
- Transferências de chamadas VoIP
- Partilha em tempo real da imagem no ecrã do utilizador
- Poking (fazer o dispositivo móvel do outro utilizador vibrar)
- Encriptação para mensagens de texto
- Troca de dispositivos durante uma conversação
- Store-and-forward (envio de mensagens quando o utilizador se encontra desligado, e que são entregues mais tarde pelo servidor quando o utilizador se ligar)
- “What you’re listening” (partilha de informação da música que o utilizador está a ouvir)

² Application Programming Interfaces

Aplicações mais vendidas

De todas as aplicações móveis existentes no mercado, algumas destacam-se pela sua visibilidade e número de utilizadores. Em seguida, caracterizam-se as aplicações que se considerou que melhor representam esse leque:

- **Skype** – www.skype.com



- Com mais de 175 milhões de utilizadores ligados por mês, o Skype disponibiliza chamadas de voz VoIP um para um, chamadas em conferência, videochamadas, instant messaging e transferência de ficheiros durante uma chamada, tudo somente ao preço utilização da ligação de dados do operador.
- Através da aquisição de créditos Skype (que já não são gratuitos), o utilizador pode realizar chamadas para números móveis que não sejam utilizadores Skype, ou para qualquer número fixo. Pode também enviar SMSs ou fazer transferência de chamadas.
- Se o utilizador adquirir uma conta Premium, obtém tarifas mais reduzidas e passa a poder participar em videochamadas em grupo (embora neste caso nos dispositivos móveis, o utilizador apenas possa usufruir da ligação áudio).

- **Fring** - www.fring.com



- Com dezenas de milhões de utilizadores e um milhão de utilizadores novos por mês, o fring disponibiliza chamadas de voz, videochamadas, videochamadas em grupo (até 4 participantes), instant messaging com integração com o MSN, ICQ, Google Talk, AIM e Yahoo Messenger, e com possível importação dos contactos dessas contas. A conversação escrita é possível decorrer durante uma chamada.
- Através da aquisição de créditos fringOut (que já não são gratuitos), podem ser feitas chamadas para qualquer número móvel ou fixo (mas apenas chamadas de voz).

- **iMessage** - www.apple.com/ios/features.html#imessage



- Uma aplicação nativa do iOS 5.
- Fornece SMSs, MMSs, e instant messaging (se detectar que o contacto também usa o iMessage, SMSs e MMSs são convertidas para instant messaging). Permite fazer transferências de ficheiros, troca de dispositivos durante uma conversa, encripta as mensagens de texto, tem notificações de entrega e de leitura, e permite também conversação em grupo.

- **Thrutu** - www.thrutu.com



- Com cerca de 250 mil downloads nos primeiros 4 meses de lançamento, e um número de downloads do Android Market entre 100 a 500 mil, o Thrutu disponibiliza um painel que é lançado durante uma chamada de onde o utilizador pode fazer partilha de ficheiros, poking (vibrar o outro contacto), partilha de localização, tirar uma fotografia com o dispositivo ou partilhar um contacto da lista de contactos.

- **Kik Messenger** – www.kik.com



- Com mais de 4 milhões de utilizadores, o Kik Messenger disponibiliza instant messaging com capacidade de store-and-forward, transferência de ficheiros, notificações de mensagens enviadas, recebidas e lidas, e conversação em grupo até 10 participantes. É um dos poucos messengers disponíveis para o Windows Phone 7.

➤ **Whatsapp** – www.whatsapp.com



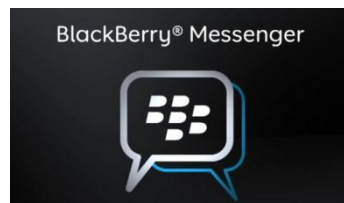
- O Whatsapp é um messenger com mais de 800 mil utilizadores apenas no Reino Unido (mais do que o Skype). A aplicação é grátis para todas as plataformas para as quais se encontra disponível, excepto para o iPhone, custando 0,99 dólares americanos na App Store da Apple.
- Disponibiliza conversação em grupo até 5 participantes, notificações de entrega das mensagens, transferência de ficheiros, informação de estado do contacto, integração com a lista de contactos nativa do dispositivo, partilha de localização, store-and-forward e imagens de fundo personalizáveis.

➤ **Viber** – www.viber.com



- Atingiu já um número de downloads no Android Market de 1 a 5 milhões.
- Permite ao utilizador efectuar chamadas de voz e instant messaging, com integração da lista de endereços nativa. As chamadas que não sejam efectuadas para utilizadores que não tenham a aplicação Viber ligada são taxadas como chamadas não-Viber.

➤ **Blackberry Messenger** – www.us.blackberry.com/apps-software/blackberrymessenger/



- Aplicação de instant messaging nativa do BlackBerry, em Setembro de 2010 tinha atingido os 28 milhões de utilizadores.
- Disponibiliza conversação em grupo até 30 participantes, notificações de entrega e de leitura das mensagens, informação da disponibilidade do contacto, partilha de ficheiros, utiliza a lista de endereços nativa, partilha de localização, pode fazer chat enquanto se encontra numa outra aplicação e também permite pking.

Comparativo das aplicações

Uma vez identificadas as principais aplicações Over-The-Top, apresentam-se dois quadros comparativos: o primeiro permite comparar as funcionalidades das aplicações, e o segundo as plataformas nas quais a aplicação pode ser instalada.

	Chamadas de Voz	Chamadas em Conferência	Videochamadas	Videochamadas em grupo	SMS	MMS	IM/chat	IM/Chat em grupo	Presença	Transferência de ficheiros	Integração com redes sociais	Partilha de localização
Skype	✓	✓	✓	✓	✓	X	✓	X	✓	✓	✓	X
fring	✓	X	✓	✓	X	X	✓	X	✓	X	X	X
iMessage	X	X	X	X	✓	✓	✓	✓	X	✓	X	✓
Thrutu	X	X	X	X	X	X	X	X	X	✓	X	✓
Kik	X	X	X	X	X	X	✓	✓	X	✓	X	X
Whatsapp	X	X	X	X	X	X	✓	✓	✓	✓	X	✓
Viber	✓	X	X	X	X	X	✓	X	X	X	X	X
BlackBerry Messenger	X	X	X	X	X	X	✓	✓	✓	✓	X	✓

Figura 1 - Comparação das aplicações OTT por funcionalidades que disponibilizam

De uma simples análise podemos concluir que nenhuma aplicação fornece todas as funcionalidades descritas, apesar de algumas estarem bastante perto disso (como é o caso do Skype). Nesta tabela encontramos já indícios da maior desvantagem das aplicações Over-The-Top no mercado: a segmentação dos clientes. Cada aplicação tem a sua base de dados de clientes, e os clientes dentro dessa aplicação só podem usufruir dos serviços com outros clientes também registados na mesma aplicação.

	Android	iPhone	iPad	iPod Touch	BlackBerry	Windows Phone 7	Nokia Symbian
Skype	✓	✓	✓	X	X	X	✓
fring	✓	✓	✓	X	X	X	✓
iMessage	X	✓	✓	✓	X	X	X
Thrutu	✓	✓	X	X	X	X	X
Kik	✓	✓	X	X	X	✓	X
Whatsapp	✓	✓	X	X	✓	X	✓
Viber	✓	✓	✓	X	X	X	X
BlackBerry Messenger	X	X	X	X	✓	X	X

Figura 2 - Comparação das aplicações OTT por plataformas para as quais estão disponíveis

Nesta tabela pode-se estender a segmentação de que se falava (a nível da aplicação) para o nível da plataforma do dispositivo móvel. Nenhuma aplicação se encontra disponível para todos os sistemas operativos, e algumas (sendo aplicações nativas) estão restringidas a um só sistema.

2.2. A resposta da GSMA

Reconhecimento das forças e fraquezas

Confrontados com a atracção dos seus clientes pelos serviços inovadores das aplicações OTT, os operadores sentiram a necessidade de descobrir uma forma de expandir os seus serviços básicos. Para obter de volta a atenção dos utilizadores, criaram um projecto que permitisse renivelar a proposta de valor dos operadores tomando partido das fraquezas das outras aplicações.

Como já foi indicado na secção anterior, as aplicações OTT têm associadas a si uma grande desvantagem: a segmentação dos clientes. Estas aplicações são baseadas em comunidades, nas quais um utilizador se deve registar para poder interagir com outros dentro da comunidade, e estão disponíveis apenas para alguns sistemas operativos. Para além disso, o serviço de dados é baseado numa política de best-effort, e a sobrevivência destas aplicações está fortemente relacionada com a visibilidade e adesão dos utilizadores (publicidade e aquisição de conteúdos/serviços in-app).

A GSMA, representante de mais de 800 operadoras telefónicas a nível mundial, apercebeu-se do enorme potencial dos operadores para superar estas fraquezas. Mais do que uns milhões de utilizadores, existe hoje um total aproximado de 5.2 mil milhões de ligações GSM e 3GSM em todo o mundo. Este mundo de comunicação é altamente interoperável, fiável e não sobrevive de receitas indirectas. Estavam reunidos os recursos que permitiam elaborar um projecto capaz de repor a relevância do operador enquanto fornecedor de comunicação. E esta reposição só pode ser feita de uma forma: aumentando o valor fornecido ao cliente.

Restituição da proposta de valor: o projecto RCE

A GSMA juntou-se aos principais fabricantes de equipamentos móveis e a empresas de desenvolvimento de aplicações para formar o RCE – Rich Communications Ecosystem - um projecto com a visão de permitir aos utilizadores realizar comunicações enriquecidas com contactos de diversas redes da sua lista de endereços, numa comunidade global interoperável. A concentração dos esforços na interoperabilidade permite a existência de implementações individuais com diferentes experiências ao nível do utilizador, algo que se revela bastante apelativo para os parceiros responsáveis pelo desenvolvimento das aplicações, e para os fabricantes dos dispositivos.

O projecto RCE não cria novos protocolos, mas define sim especificações que utilizam serviços já utilizados pelos subscritores de banda larga. Foram propostas várias versões da especificação até se ter chegado ao compromisso de uma só, com o intuito de obter uma rápida entrada no mercado. Essa especificação final tem o nome de RCS-e – Rich Communications Suite enhanced. [10]

2.3 A especificação RCS-e / Joyn

Funcionalidades

A especificação RCS-e tem como objectivo fornecer uma extensão simples e interoperável aos serviços de voz e texto disponibilizados pelos operadores. Esta extensão funcionará sobre o serviço de dados dos operadores, e foi desenhada em vista a expandir o leque de serviços adicionais ao longo do tempo. Inicialmente acrescentará à pilha de comunicações sobre IP os serviços de:

- Instant Messaging
- Transferência de ficheiros
- Partilha de imagens e de vídeo (durante uma chamada)

O que estes serviços (e os futuros serviços) têm em comum é o facto de assentarem sobre um mecanismo fundamental de descoberta das capacidades de um contacto. O objectivo é de que o utilizador disponha de uma actualização em tempo real (ou o mais próximo possível disso) das interações que pode iniciar com um contacto.

Com vista a acelerar a entrada no mercado, para uma aplicação obedecer à especificação basta disponibilizar: os procedimentos para provisionamento e registo na rede, o mecanismo de descoberta de capacidades e a funcionalidade de Instant Messaging. Os restantes serviços são inicialmente opcionais. O objectivo é permitir que mesmo os dispositivos mais limitados possam participar em comunicações RCS-e. No entanto, os serviços disponibilizados devem estar sempre de acordo com as capacidades de hardware do dispositivo, apresentando-se disponíveis sempre que possível.

Pretende-se que estes serviços no futuro estejam nativamente integrados nos dispositivos, mas numa primeira fase comercial, os serviços serão disponibilizados via aplicações para smartphones e tablets.

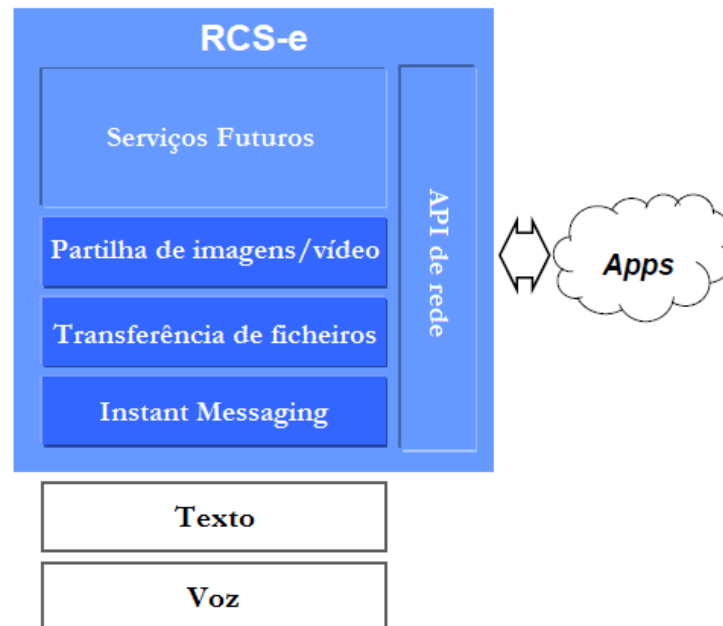


Figura 3 - Pilha de serviços propostos pela especificação RCS-e

A arquitectura IMS

A especificação RCS-e faz uso da arquitectura de rede IMS (IP Multimedia Subsystem)[15], uma arquitectura utilizada pelos operadores de telecomunicações para integrar serviços fornecidos pela rede IP com os serviços fornecidos pelos dispositivos móveis digitais (convergindo-os numa só infra-estrutura IP).

Apesar da infra-estrutura de rede celular permitir o acesso aos serviços de Internet, a utilização da arquitectura IMS traz vantagens adicionais, tais como:

- Representa uma plataforma comum standardizada com componentes reutilizáveis, permitindo reduzir o tempo de entrada no mercado de um serviço
- Fornece serviços multimédia com Quality of Service, em detrimento de uma política “Best-effort”
- Permite aos operadores cobrarem pela qualidade do serviço e não por bytes transferidos
- Disponibiliza serviços independentemente da localização do utilizador (ultrapassando os obstáculos de roaming).

Na página seguinte podemos visualizar um diagrama da arquitectura IMS.

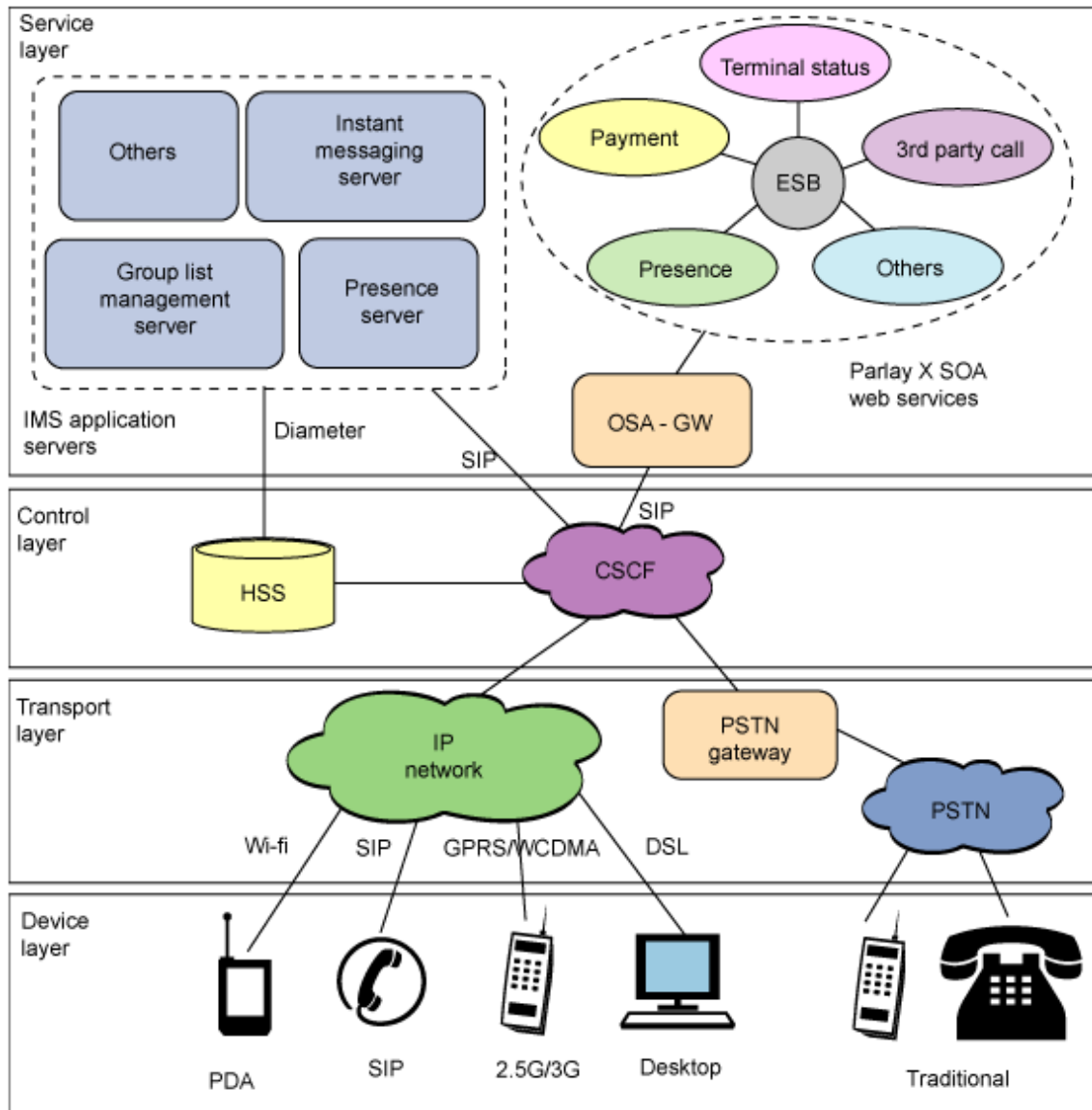


Figura 4 - Arquitetura IMS (IP Multimedia Subsystem)

A arquitetura IMS faz uso extensivo do protocolo SIP (Session Initiation Protocol) [13] do qual falamos em seguida. A restante análise em detalhe da arquitetura está fora do âmbito deste estágio.

O protocolo SIP

O protocolo SIP é utilizado para criar, modificar e terminar sessões com um ou mais participantes. A um nível elevado, entenda-se o conceito de sessão como uma troca de informação entre uma associação de participantes. É um protocolo de controlo usado para a comunicação sobre IP e é responsável pela autenticação dos utilizadores, gestão de chamadas, partilhas de conteúdo, descoberta de serviços activos, e muitos outros casos de uso.

Funciona sobre vários protocolos de transporte, mas sem qualquer dependência deles, e assume a existência de um servidor que permita registar um agente, e reencaminhar as suas mensagens para outros agentes.

A simplicidade deste protocolo torna-o escalável, extensível e polivalente, qualidades extremamente importantes para os serviços RCS-e. No diagrama de sequência abaixo podemos visualizar um exemplo de troca de mensagens SIP no caso de uso de registo de um utilizador no SIP Server de um operador. Por baixo encontra-se um exemplo de uma mensagem SIP.

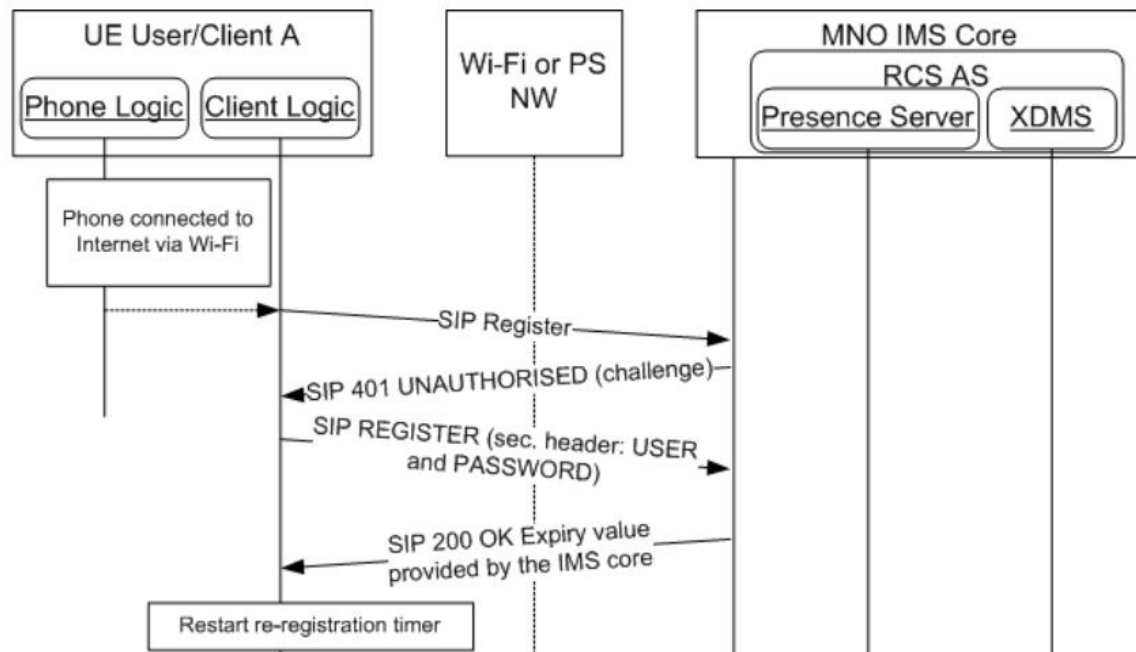


Figura 5 - Exemplo da sequência de troca de mensagens SIP no registo de um utilizador

```

REGISTER sip:registrar.biloxi.com SIP/2.0
Via: SIP/2.0/UDP bobspc.biloxi.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Bob <sip:bob@biloxi.com>;tag=456248
Call-ID: 843817637684230@998sdasdh09
CSeq: 1826 REGISTER
Contact: <sip:bob@192.0.2.4>
Expires: 7200
Content-Length: 0
  
```

O protocolo MSRP

Para efeitos de instant messaging, transferência de ficheiros e partilha de imagens, a especificação RCSe utiliza o protocolo MSRP (Message Session Relay Protocol) [14].

O MSRP trabalha no contexto de uma sessão de comunicação (estabelecida pelo protocolo SIP), e funciona somente sobre o protocolo de transporte TCP. Isto significa que quando sobre UDP, os casos de uso mencionados acima não podem ser implementados com o uso de mensagens MSRP. Nesses casos, as mensagens utilizadas são as mensagens SIP MESSAGE do protocolo SIP:

Lançamentos

Os 5 maiores operadores móveis da Europa (Deutsche Telekom, Orange, Telecom Italia, Telefonica e Vodafone) comprometeram-se [11] a lançar serviços de comunicação RCS-e na Europa e na Ásia durante este ano de 2012. A Telefonica foi já a primeira. Outros lançamentos estão previstos para a América do Norte, e muitos outros encontram-se correntemente em discussão.

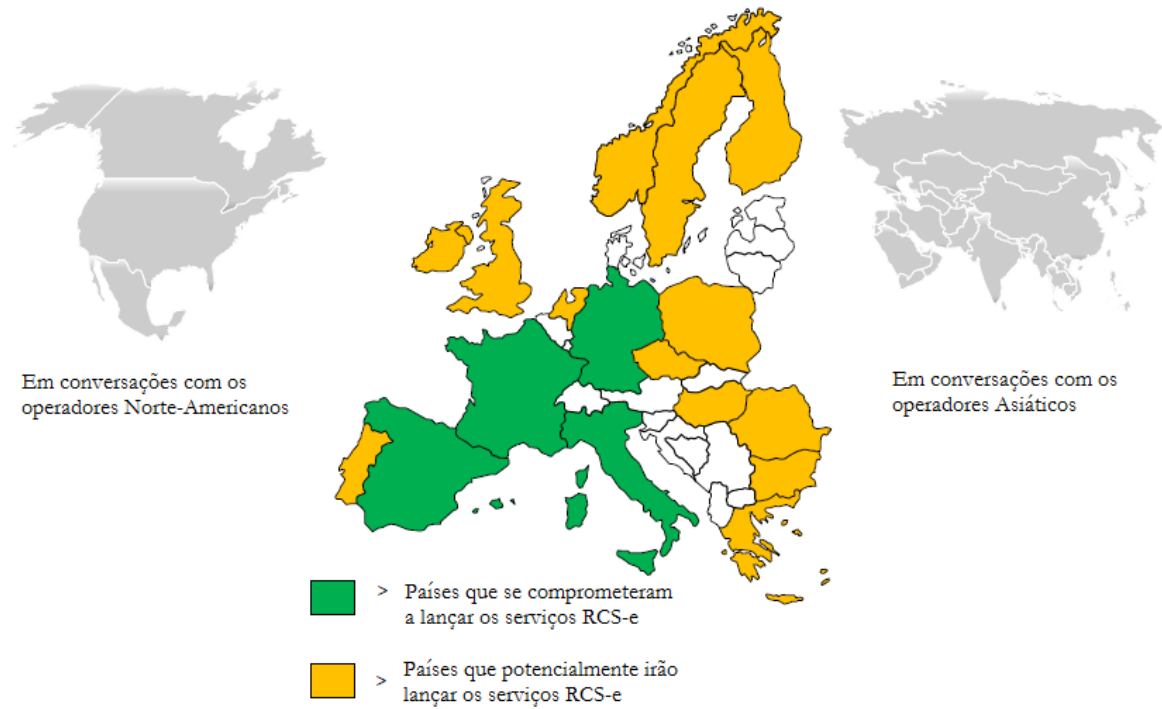


Figura 6 - Compromisso de diferentes países em lançar o RCS-e em 2012

Capítulo 3

Objectivos do Estágio e Método de Abordagem

A suite de produtos Wit Communications Suite [12] (WCS) representa a oferta da Wit Software aos operadores móveis que procuram uma solução RCS. A suite inclui um conjunto de aplicações cliente para várias plataformas, sendo que uma delas em específico - o WIT Mobile Communicator (WMC) – encontra-se já disponível para Android (embora ainda só tenha sido lançada em Espanha), e estará brevemente disponível para iPhone e iPad.

Este estágio dedica-se ao desenvolvimento da aplicação WMC para os dois dispositivos da Apple: o iPhone e o iPad, sendo que para o iPhone o estagiário foi integrado no decorrer dos desenvolvimentos, ficando responsável por apenas algumas funcionalidades, e para o iPad iniciou o desenvolvimento da aplicação de raiz juntamente com um outro colaborador da empresa.

3.1 A WIT Communications Suite

A WCS foi lançada com sucesso no mercado em 2007 e, desde então, tem sido instalada em vários operadores por todo o mundo. A suite tem uma grande importância na oferta da WIT Software e está a ser um pilar importante para a expansão da empresa no estrangeiro. Disponibiliza clientes para o PC (WPC), Web (WWC) e dispositivos móveis (WMC), assim como uma solução de servidor (WCAS).

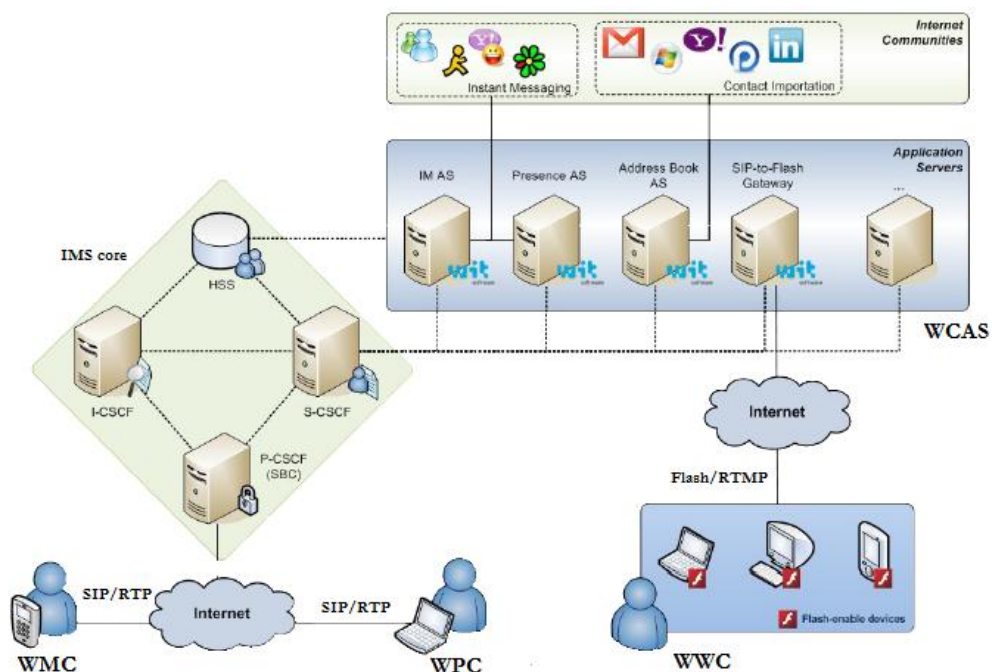


Figura 7 - Arquitectura global da WIT Communications Suite

3.2 A aplicação WMC



Independentemente do dispositivo móvel, todas as versões da aplicação WMC pretendem disponibilizar as mesmas funcionalidades e serviços (excepto casos em que o sistema operativo do dispositivo não o permita):

- Gestão de contactos (incluindo sincronização com listas nativas ou com redes sociais)
- Chamadas de voz (1 para 1 e em conferência)
- Videochamadas
- Partilha de imagens e vídeo durante uma chamada
- Chat/Instant Messaging
- Transferência de ficheiros
- Partilha de localização
- Descoberta das capacidades do contacto em tempo real

A WIT desenvolveu uma biblioteca de comunicação – a **Communicator Library** – com o intuito de ser partilhada por todos os clientes e permitir ao máximo isolar a programação específica ao dispositivo (programação em Objective C para os dispositivos iOS, e programação em Java para os dispositivos Android).

Esta biblioteca está em constante evolução, geralmente para responder a alterações nas especificações, e faz uso de um conjunto de bibliotecas externas (umas desenvolvidas pela wit e outras open-source ou cujas licenças foram adquiridas).

Na próxima página é ilustrada a arquitectura comum à aplicação WMC iPhone e WMC iPad, localizando a commlib e descrevendo as bibliotecas mais utilizadas e a utilidade das mesmas na aplicação.

Arquitetura dos clientes

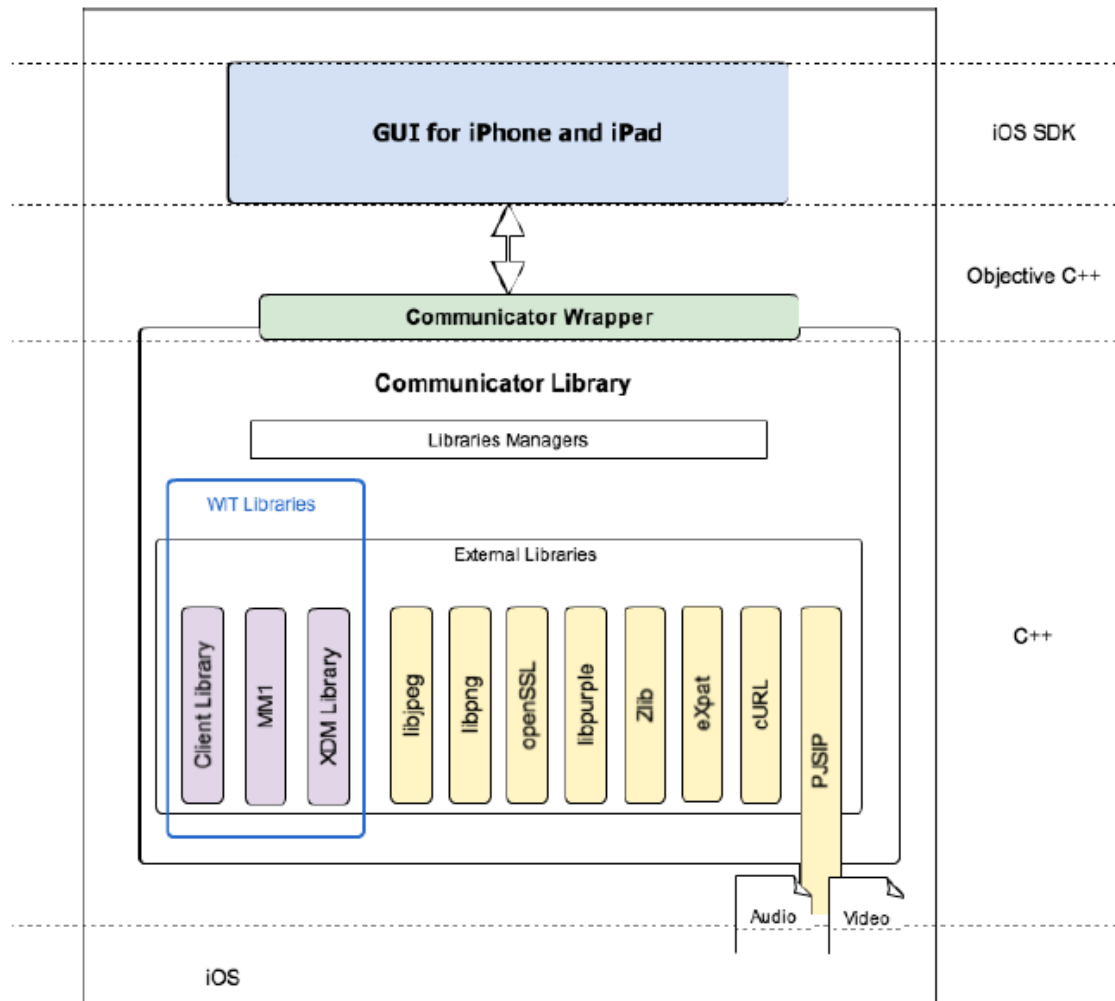


Figura 8 - Diagrama de arquitectura dos clientes WMC iPad e WMC iPhone

Sobre a camada nativa de iOS, a Communicator Library implementada em C++ um conjunto de Managers que abstraem o uso das bibliotecas externas. Estes Managers interagem com a camada acima, implementada em Objective C++, que acaba por funcionar como wrapper com o intuito de separar por completo o código de Objective C (código que utiliza o SDK iOS, que é o responsável pela lógica de UI) da restante lógica da Communicator Library.

Das bibliotecas externas, destacam-se como sendo as duas mais importantes:

- **PJSIP** – É a biblioteca de comunicação que representa a fundação de toda a comunicação SIP, suportando protocolos como o RTP e SDP, e disponibilizando mecanismos para atravessar NAT. Suporta audio e video, mas também presença e instant messaging. Apesar de existirem outras soluções no mercado com o Osip [16], OpenSipStack [17] e reSIPProcate [18], o PJSIP distingue-se das restantes pela sua portabilidade e fiabilidade no suporte em várias plataformas móveis.
- **Client Library** – Para ultrapassar obstáculos na implementações em C/C++ nas diferentes plataformas, a WIT criou a biblioteca (para uso interno) Client Library. Esta biblioteca fornece abstrações para funções de baixo nível, nomeadamente na implementação de threads, listas, hash-tables, persistência de dados, gestão de ficheiros, e gestão de memória.

3.3 Objectivos para o WMC iPhone

Após ser feito um estudo do estado da arte, os objectivos para o primeiro semestre neste estágio começaram por passar por uma ambientação à programação para iOS e ferramentas de desenvolvimento e uma familiarização com a Communicator Library (incluindo algumas das bibliotecas utilizadas pela CommLib).

De seguida iniciaram-se os desenvolvimentos, e foi entregue ao estagiário a responsabilidade de desenvolvimento das seguintes funcionalidades:

- **Partilha de imagem durante uma chamada** – Ao estabelecer uma chamada com um contacto RCSe, se esse contacto tiver a capacidade de efectuar uma partilha de imagem, o utilizador deve ser alertado dessa possibilidade, permitindo-lhe regressar à aplicação e escolher uma imagem da sua galeria para iniciar a partilha. O contacto que atende a chamada poderá a qualquer altura regressar à aplicação (com a chamada a decorrer) e efectuar uma partilha de imagem através da lista de contactos. A especificação RCSe define também a existência de limites de tamanho máximo e limites de aviso (que não impossibilitam a transferência mas devem criar um alerta para o utilizador). Estes limites são normalmente definidos pela configuração do operador do utilizador.
- **Background Multitasking** – o sistema operativo iOS não permite que uma aplicação permaneça activa em modo background por mais de 10 minutos, com a excepção de aplicações VoIP. Para tomar partido dessa excepção, as aplicações VoIP devem criar sockets activos que mantenham a ligação aberta e impeçam o iOS de a terminar.
- **Protecção do estado desligado dos serviços de localização** – para uma aplicação para iPhone aceder às imagens da galeria, deve ser criada uma protecção que lide com o facto dos serviços de localização estarem desligados. O problema advém de algumas fotografias poderem ter geotagging³ associado, o que faz o sistema operativo impedir o acesso às mesmas no caso dos serviços de localização estarem desligados.
- **Edição do avatar na vista de um contacto** – ao seleccionar um contacto, é aberta uma vista com os detalhes e opções de comunicação com o mesmo. Um dos detalhes corresponde ao avatar (imagem) do mesmo. Esse avatar deve poder ser inserido, substituído ou removido, e deve conter uma imagem pré-definida na ausência de um.
- **Remoção da barra de estado da chamada** – Durante uma partilha de imagem em chamada, no caso da chamada terminar gerava-se uma incoerência nas vistas da aplicação, mantendo-se a barra verde de chamada mesmo tendo ela já terminado.
- **Cancelamento do Sign-In** - Numa fase de comercialização a identificação do utilizador será feita automaticamente pelo MSISDN⁴ do cartão SIM. No entanto, na fase de desenvolvimentos era usado um ecrã de login, a partir do qual o utilizador não conseguia cancelar o processo de registo uma vez iniciado. Este cancelamento deveria manter a estabilidade da aplicação, e interromper o início dos serviços sem efeitos indesejados.
- **Optimização do processo de Sign-Out** – Inicialmente o processo de Sign-Out era muito longo (cerca de 17-18 segundos), e foi necessário fazer uma análise que permitisse otimizar a duração do mesmo.

³ Geotagging – Adição de metadados geográficos à fotografia (localização de onde foi capturada)

⁴ MSISDN - Número que identifica unicamente o cartão SIM

- **Histórico de comunicação por contacto** – Num projecto em paralelo com o dos outros desenvolvimentos (referente à mesma aplicação, mas noutra versão), foi necessária a inclusão do histórico de comunicações com um contacto. Este histórico permitia ver registos de chamadas, conversações passadas e registos de transferências de ficheiros.
- **Suporte para o Airplane Mode** – A aplicação ficava bastante instável quando iniciava ou transitava para um modo offline (por perda de ligação ou por activação do modo “Vôo”. Foi preciso adicionar o suporte que lidasse com essas situações.

3.4 Objectivos para o WMC iPad

Para o segundo semestre foi planeado o desenvolvimento da aplicação para iPad de raiz. A aplicação iria também tomar partido da Communicator Library, e deveria disponibilizar o mesmo leque de serviços que as outras aplicações WMC dos outros dispositivos móveis, salvo algumas excepções.

Funcionalidades inicialmente previstas

As funcionalidades planeadas seriam então:

- Lista e Gestão de contactos
- Chamadas de voz e de vídeo
- Chat/Instant Messaging
- Envio e recepção de SMS e MMS
- Gestão de chamadas
- Transferência de ficheiros
- Partilha de conteúdos durante uma chamada
- Integração com redes sociais
- Localização dos contactos num mapa

No entanto, a empresa teve a necessidade de adiar o desenvolvimento do iPad para mais tarde no semestre, e realocou o estagiário para um projecto iPhone muito semelhante ao do primeiro semestre, atribuindo-lhe a responsabilidade da implementação de algumas funcionalidades.

Realocação para o WMC iPhone RCS2

O projecto para o qual o estagiário foi realocado seguia uma especificação muito semelhante à do RCSe: o RCS Release 2. As principais diferenças entre as duas especificações são:

- O RCSe não faz uso do módulo de presença, enquanto que o RCS2 faz
- O RCS2 disponibiliza chamadas de VoIP e videochamadas, ao contrário do RCSe.
- O RCSe utiliza os contactos nativos do telefone/iPad, e o RCS2 guarda os contactos do utilizador num documento XCAP que é acedido remotamente ao iniciar a aplicação.

Neste projecto, o estagiário teve a seu cargo as seguintes tarefas:

- Implementação da funcionalidade de Image Share sobre uma chamada de VoIP.
- Implementação da funcionalidade de Video Share sobre uma chamada de VoIP.
- Identificação e correcção de problemas encontrados na aplicação

- Algumas alterações à User Interface.

Regresso ao WMC iPad

Mais tarde no semestre foi então iniciado o desenvolvimento do WMC iPad. Os requisitos funcionais mantiveram-se, mas como o tempo de estágio restante não era suficiente para a implementação de todas as funcionalidades, foram incumbidas ao estagiário (e ao colaborador que o acompanhou no projecto iPad) as seguintes tarefas:

- Definição da arquitectura da aplicação
- Implementação do esqueleto de User Interface da aplicação
- Implementação do módulo de Sessão
- Implementação do módulo de Contactos
- Implementação do módulo de Capabilities
- Implementação do módulo de Chat

A implementação do último módulo seria uma tarefa longa e ultrapassaria a data limite do estágio, pelo que o estagiário ficou apenas responsável pela criação da User Interface e implementação do esqueleto da bridge/wrapper no módulo de chat.

3.5 Metodologia de desenvolvimento

Para o desenvolvimento da aplicação para iPhone e da aplicação para iPad, foram consideradas duas metodologias a seguir: uma metodologia em cascata (Waterfall) [19] ou uma metodologia Agile [20].

Metodologia Waterfall

Possivelmente uma das mais antigas metodologias de desenvolvimento, a metodologia Waterfall promove um desenvolvimento de software sequencial, dividindo-o numa série de fases consecutivas com pontos de início e fim bem definidos.

O progresso é medido consoante as fases que já foram atravessadas, e no máximo permite-se que se retroceda uma fase para efectuar modificações (motivadas, por exemplo, por alterações nos requisitos, ou descoberta de problemas). Tipicamente as fases atravessadas consistem nas seguintes:

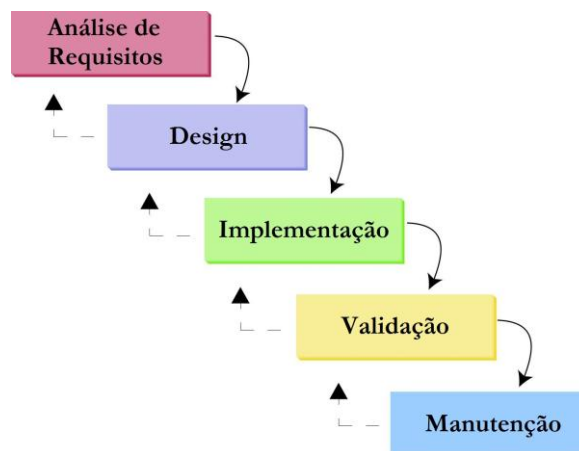


Figura 9 - Diagrama de fluxo da Metodologia Waterfall

- **Análise de requisitos**
 - Definir com clareza os objectivos do cliente e perceber o contexto e o problema para o qual se procura uma solução são provavelmente as tarefas mais importantes nesta metodologia. As funcionalidades que o software deve disponibilizar, os sistemas com que deve interagir e os níveis de desempenho a que o deve fazer, são geralmente especificados em documentos de requisitos formais.
- **Design**
 - Esta fase pretende definir o esqueleto da solução que irá ser implementada para satisfazer os requisitos identificados. A elaboração da arquitectura do software/hardware, dos modelos de dados, das linguagens e ferramentas utilizadas, entre outras escolhas, são geralmente reunidas em documentos de especificação de design.
- **Implementação**
 - A codificação das componentes da(s) arquitectura(s) é feita pela equipa de desenvolvimento, e deve ser acompanhada com o uso de ferramentas de controlo de versões quando as equipas têm grandes dimensões (permitindo descartar alterações indesejadas no caso de se encontrarem problemas). Também a equipa de design realiza o trabalho gráfico de interface com o utilizador, a par do trabalho dos restantes especialistas da equipa de desenvolvimento (som, vídeo...).
- **Validação**
 - Nesta fase é feita a verificação do cumprimento dos requisitos identificados na primeira fase. Procuram-se os erros, identificam-se, e reportam-se para correcção pela equipa de desenvolvimento. Os testes podem ser realizados por uma equipa de testes ligada ao produto, ou podem tomar a forma de testes de aceitação (grupos de testes que são construídos normalmente pela mão do próprio cliente). Quando o produto se considera apto para lançamento, é entregue ao cliente, e é feita a instalação/fornecimento da versão actual.
- **Manutenção**
 - Já com o software lançado, é comum (senão inevitável) que sejam identificados erros ao longo do tempo, ou mesmo alterações aos requisitos por parte do cliente. A equipa de manutenção deve estar pronta pra realizar as modificações necessárias, gerando novas versões à medida que estas modificações são introduzidas.

Metodologia Agile

A metodologia de desenvolvimento Agile surgiu devido à necessidade de existência de uma metodologia adaptada à realidade do mercado de software.

Ao contrário das metodologias tradicionais que tendem a abordar os requisitos de forma rígida e definitiva, a Agile aceita a realidade de que os requisitos, análise e design, nunca chegam a uma fase de término. Na verdade, dada a competitividade empresarial e o leque crescente de soluções de software no mercado, são poucos os projectos em que os requisitos não mudam constantemente. Para as empresas, aceitar esta realidade pode significar uma redução grande nos custos de desenvolvimento.

Para responder da melhor forma aos requisitos emergentes a Agile transforma o cliente num membro influente e essencial da equipa, permitindo-lhe tomar decisões ao longo do projecto. A geração de versões tem lugar desde o início do projecto, e não apenas na fase final, o que permite ter algo para mostrar ao cliente com maior frequência, e validar mais cedo se a implementação satisfaz as suas necessidades. O progresso passa então a ser medido pelo software que está funcional num dado momento.

Também a gestão de actividades inicialmente planeada é substituída por uma gestão de requisitos. O desenvolvimento é feito do requisito/funcionalidade mais importante até à menos importante. Isto permite não só parar num dado momento e ter as funcionalidades mais importantes já feitas, como também obter estimativas de custos por funcionalidade em vez de custos por actividade (se por exemplo, os fundos do projecto se esgotarem, não teria sido gasto dinheiro em análise e design de funcionalidades que não serão implementadas).

De uma forma resumida, o manifesto da metodologia Agile valoriza:

- **Indivíduos e interacções** em detrimento de processos e ferramentas
- **Software funcional** em detrimento da documentação compreensiva
- **Colaboração com o cliente** em detrimento da negociação de contracto.
- **Resposta à mudança** em detrimento de seguimento de um plano.

Tipicamente a metodologia segue o fluxo apresentado abaixo:

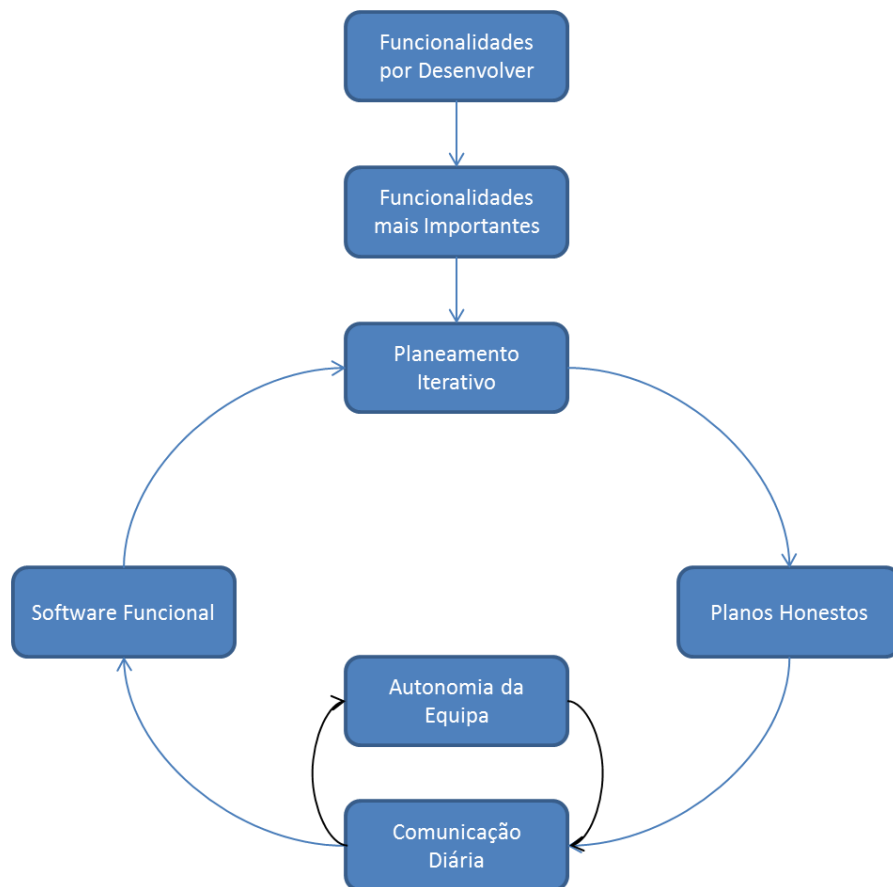


Figura 10 - Diagrama de fluxo da Metodologia Agile

As fases atravessadas são então as seguintes:

- Funcionalidades por Desenvolver
 - Inicialmente deve ser constituída uma lista das funcionalidades do produto pretendido. Esta lista deve ser revista regularmente de forma a adicionar ou remover requisitos do cliente.
- Funcionalidades mais Importantes
 - Da lista acima devem ser escolhidas as funcionalidades mais importantes. A importância é relativa, mas deve ser bem definida para o projecto. Alguns factores de avaliação da importância podem ser: o retorno de investimento da funcionalidade, as dependências dentro da arquitectura do sistema, o grau de utilização pelo consumidor final, etc. Regra geral é considerada uma combinação destes factores.
 - As estimativas de esforço nesta fase devem ser muito generalistas uma vez que muitas destas funcionalidades podem não vir sequer a serem implementadas.
- Planeamento Iterativo
 - O planeamento iterativo consiste na realização de um plano, implementação do software, e reajustamento do plano de acordo com o que foi aprendido. Esta modificação do plano tem como principal fonte de informação o feedback das entregas incrementais de software funcional (uma entrega ao final de cada iteração).
 - Pequenas iterações dão a oportunidade de reajustar mais vezes o plano.
- Planos Honestos
 - Mais do que estimativas feitas com bom senso, os planos honestos aceitam a realidade de que existem erros honestos, e que um plano tem um prazo de expiração igual ao seu tempo de vida realmente útil. Como estamos a planear uma lista de funcionalidades juntamente com o cliente, as decisões/pormenores técnicos devem ser completamente abstraídos.
- Autonomia da Equipa
 - A não ser em casos onde existe apenas uma pessoa de confiança para tomar decisões, se a equipa for constituída por membros competentes, atribuir-lhes poder de decisão aumenta a eficiência e explora o potencial da equipa. Mantém-se um poder centralizado responsável pela marcação de reuniões de planeamento e alocação de recursos, mas é distribuído um poder de negociação e de avaliação de obstáculos pelo resto da equipa. Uma pessoa fica mais envolvida numa decisão quando fez parte dela.
- Comunicação Diária
 - A equipa deve reunir-se todos os dias, preferencialmente num espaço que promova a comunicação entre todos os membros de todas as equipas, incluindo o cliente. Esta comunicação é essencial para a autonomia da equipa.
- Software Funcional
 - Só ao fim de testes unitários e testes de qualidade de uma versão do software é que se pode dizer que temos software funcional. Estas demonstrações são extremamente importantes nas situações em que o cliente não está muito esclarecido das soluções que pode/pretende ter para os seus requisitos.

A metodologia escolhida

Considerando as duas hipóteses de metodologias a seguir, foi escolhida a metodologia Agile.

A metodologia Waterfall foi descartada devido à natureza dos projectos da aplicação iPhone e iPad. Foi previsto (e verificado) que estas aplicações iriam sofrer inúmeras alterações aos seus requisitos pelos seguintes motivos: Pelo facto das especificações seguidas estarem em constante redefinição; e devido ao conhecimento extraído do histórico de interações da empresa com o tipo de clientes para estas aplicações (a dimensão do investimento efectuado pelos operadores móveis no lançamento de uma aplicação obriga-os a efectuar imensos testes de aceitação, o que por muitas vezes produz feedback que leva a alteração de requisitos. Também a competitividade diária característica destes clientes produz requisitos emergentes de funcionalidades inovadoras capazes de superar as dos seus competidores).

Sabendo isto à partida, a metodologia Agile adaptava-se muito bem ao desenvolvimento que se tinha pela frente. Dado que as aplicações continuam em desenvolvimento, ainda hoje os clientes participam activamente (embora via telefónica/video-conferência) nas tomadas de decisões, e nas validações das versões geradas. Também a equipa de desenvolvimento tem voz activa na tomada de decisões, especialmente nos momentos em que se avalia a exequibilidade da implementação de uma dada funcionalidade.

3.6 Ferramentas utilizadas

Ao longo do estágio, o estagiário teve à sua disposição várias ferramentas para o desenvolvimento da aplicação. Além das ferramentas imprescindíveis a nível de hardware (um computador Mac para desenvolvimento em iOS, iPhones e um iPad para testar a aplicação e as funcionalidades desenvolvidas, etc), o estagiário utilizou as seguintes aplicações:

- Xcode [21] – O Xcode é um IDE inserido na package de ferramentas de desenvolvimento disponibilizada pela Apple. É o mesmo IDE que a própria Apple utiliza para desenvolver aplicações para OS X (o sistema operativo dos computadores Mac) e iOS. A package contém também ferramentas úteis como:
 - Interface Builder – Uma ferramenta de design integrada no Xcode que permite uma criação de vistas de forma muito intuitiva, assim como a associação de código aos eventos gerados pelas componentes de UI.
 - Instruments – Uma ferramenta de análise extremamente útil para detectar problemas de desempenho de uma aplicação (como memory leaks ou usos anormais do processador), ilustrando-os graficamente em tempo real.
 - Compilador LLVM – O compilador mais recente da Apple tem uma forte ligação à indexação do IDE, possibilitando o “auto-complete” do código com elevada eficiência, lançando avisos ou erros durante a escrita do código, e sugerindo correcções com grande precisão.
 - FileMerge – Um utilitário muito intuitivo para comparar ficheiros, permitindo realizar fusões de código de uma forma simples e rápida.
 - iOS Simulator – Para os momentos em que por qualquer motivo não haviam dispositivos disponiveis para teste, o iOS Simulator (iPhone e iPad) permitia efectuar todo o tipo de testes, à excepção dos que envolvessem chamadas nativas.

- Wireshark [22] – Quando uma interação com a aplicação não gera qualquer comportamento, a primeira análise a fazer é a localização do ponto de falha. Para esse efeito, o Wireshark tornou-se uma ferramenta imprescindível. Este analisador de protocolos de rede permitiu capturar em tempo real as trocas de mensagens SIP, MSRP, RTP ou mesmo HTTP, analisar o seu conteúdo, e perceber se um ponto de falha se encontrava no cliente, no servidor, ou na própria rede.

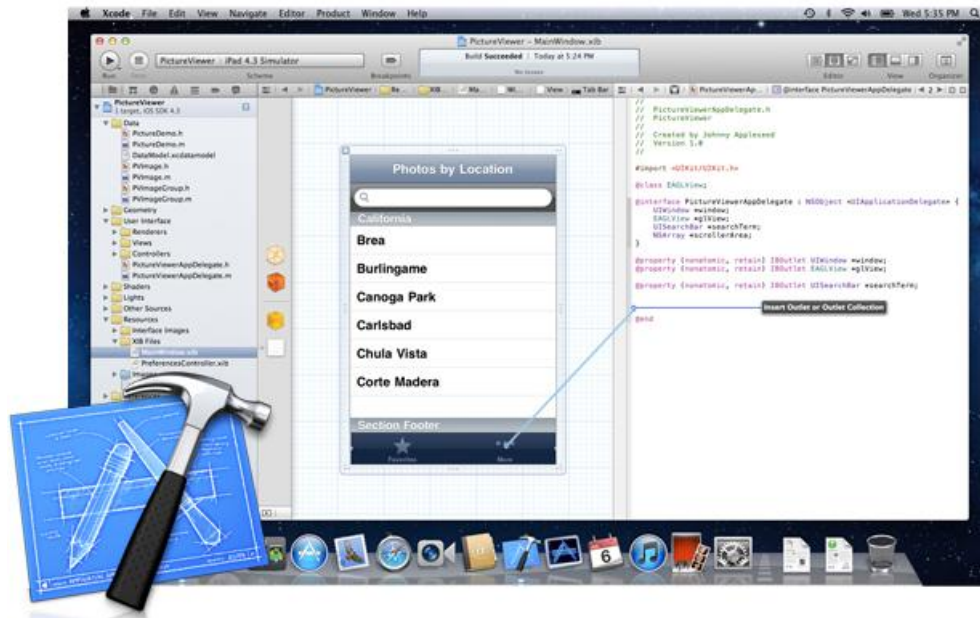


Figura 11 - IDE Xcode, com o Interface Builder aberto

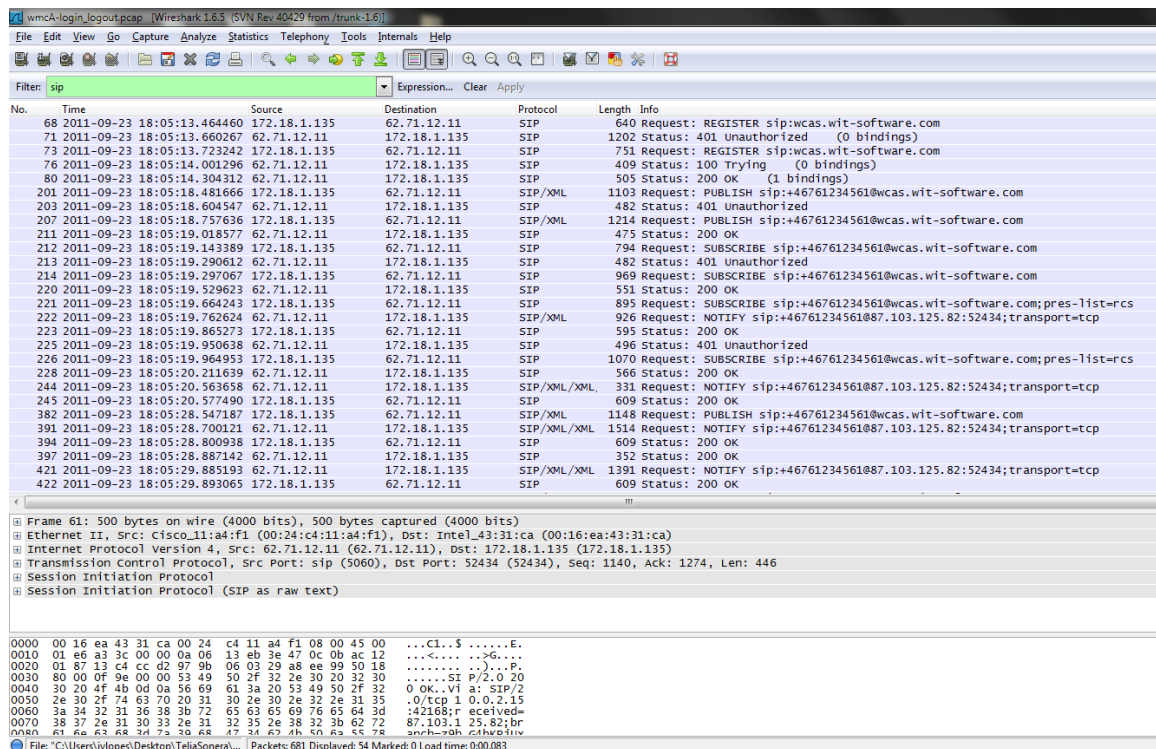


Figura 12 - Wireshark numa captura utilizando um filtro "sip"

- svnX [23] – Esta aplicação disponibiliza uma interface gráfica para um cliente de controlo de versões. Permite navegar nas directorias, pesquisar logs, fazer comparativos de versões, obter o conteúdo de um repositório, verificar quem inseriu uma dada linha de código, e tudo de forma intuitiva e drag-and-drop.

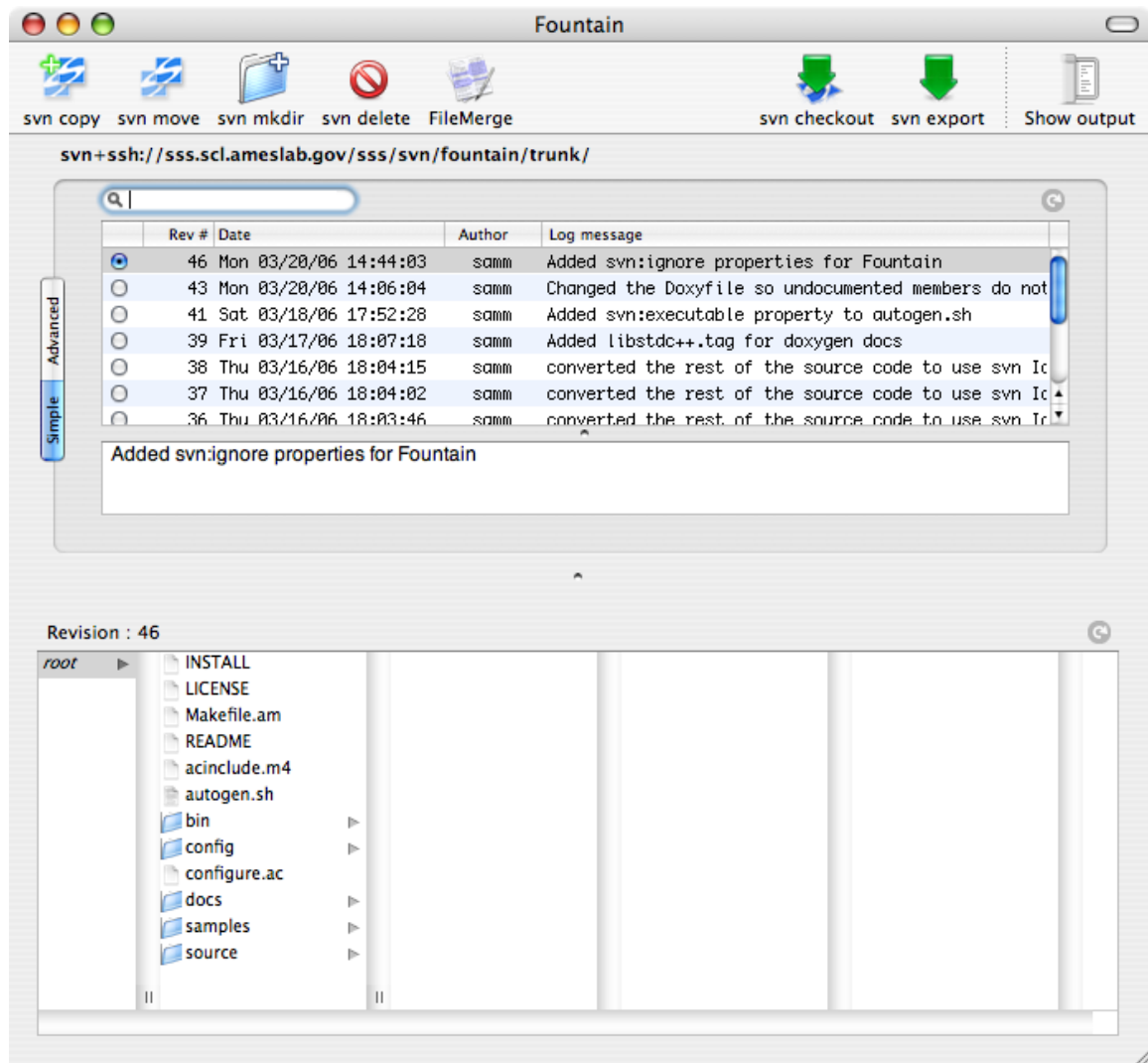


Figura 13 - Visualização de um repositório no svnX

- Mantis [24] – Para registo e manutenção dos bugs de uma aplicação utilizámos o sistema gratuito Mantis. Foi escrito em PHP, e permite que qualquer web-browser sirva como cliente. Disponibiliza funcionalidades como classificação de prioridades, atribuição a projectos, associação de comentários, e muito mais.

Capítulo 4

Plano de Trabalho e Implicações

O plano deste estágio sofreu algumas alterações em relação ao plano inicialmente proposto. Na proposta de estágio da empresa ao Departamento de Engenharia Informática da Universidade de Coimbra, o estagiário iria levar a cabo o desenvolvimento de uma aplicação para a plataforma iPad no primeiro semestre, e para tablets Android no segundo semestre.

No entanto, a empresa recebeu um RFP (Request for Proposal) da GSMA, requisitando uma proposta de desenvolvimentos para Android e iPhone. Uma vez que a programação para iPhone e iPad é muito semelhante, e dado que os desenvolvimentos em iPad constituíam uma grande componente do estágio, foi decidido que o estagiário seria integrado na equipa de desenvolvimentos do iPhone.

Assim, o estágio passou a dedicar-se à obtenção de um conhecimento aprofundado na programação em iOS, fornecendo ao estagiário a experiência de desenvolvimento de aplicações não só desenhadas para tablets, como também para smartphones.

No segundo semestre, o plano foi de novo modificado devido à necessidade da empresa de realocar temporariamente recursos para outro projecto. O estagiário foi então mudado para um projecto muito semelhante ao do primeiro semestre, onde foi encarregue de implementar algumas funcionalidades. No entanto, ao fim de 2 meses retomou o rumo do desenvolvimento da aplicação para iPad, o que seguiu até à conclusão do estágio.

Nas próximas páginas encontram-se os diagramas de Gantt referentes ao planeamento inicial, e ao planeamento após as modificações.

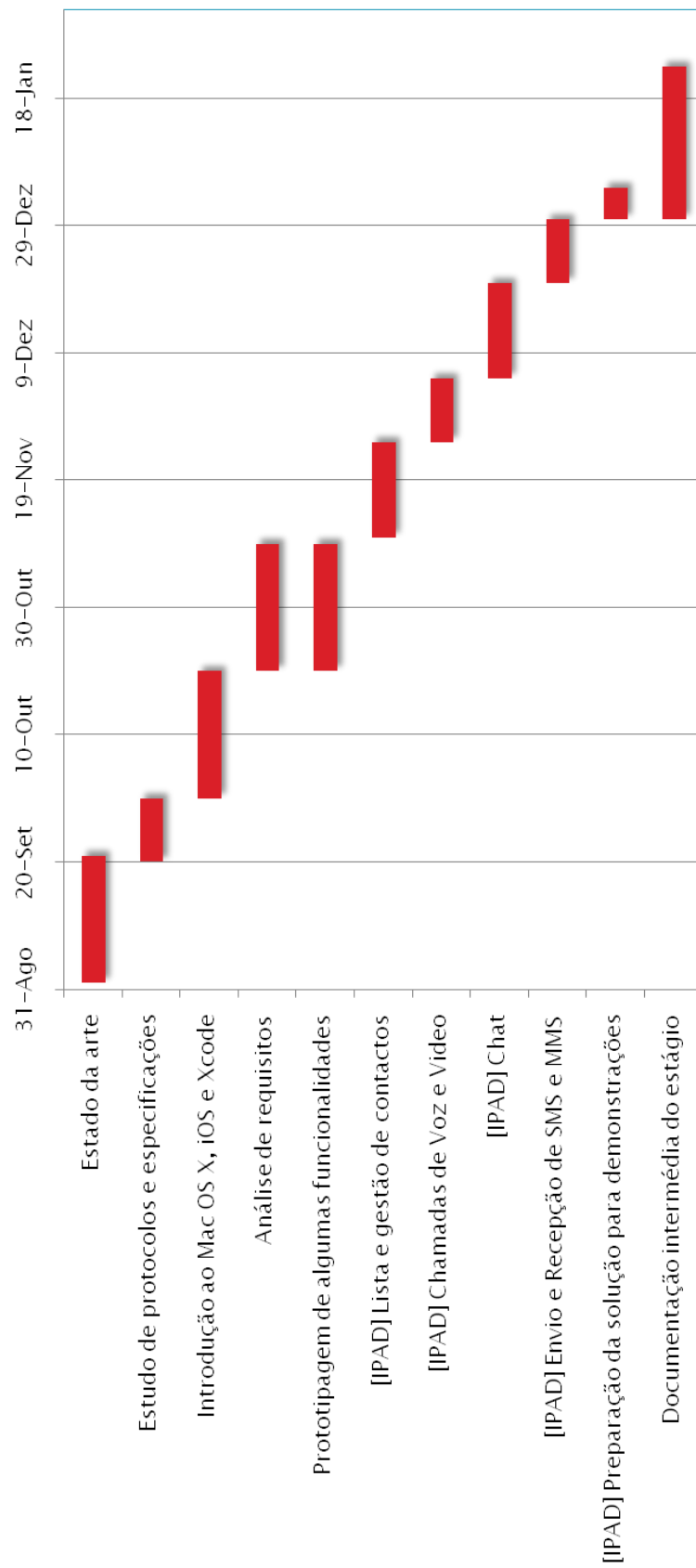


Figura 14 - Planeamento inicial para o 1º semestre

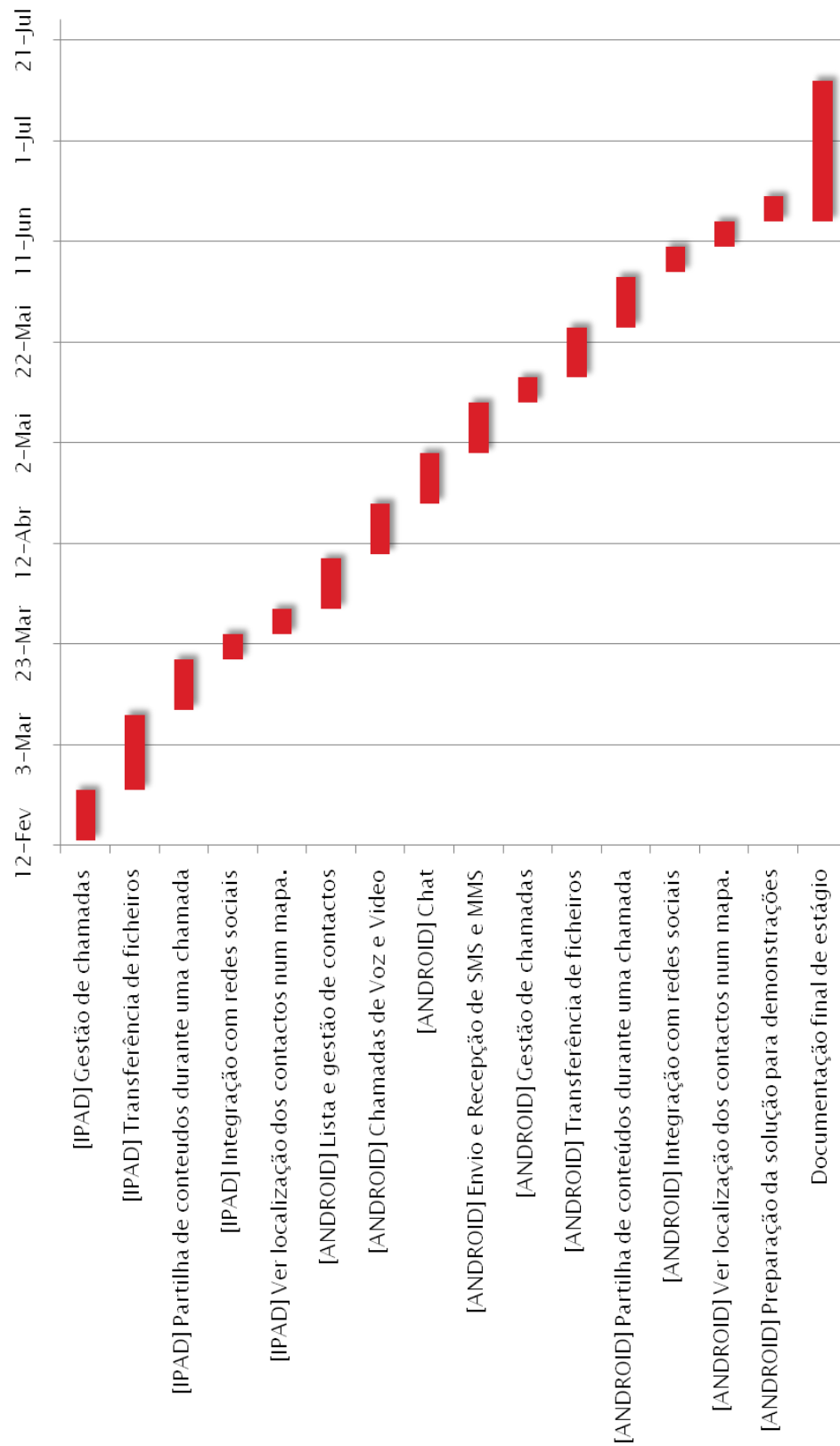


Figura 15 - Planeamento inicial para o 2º semestre

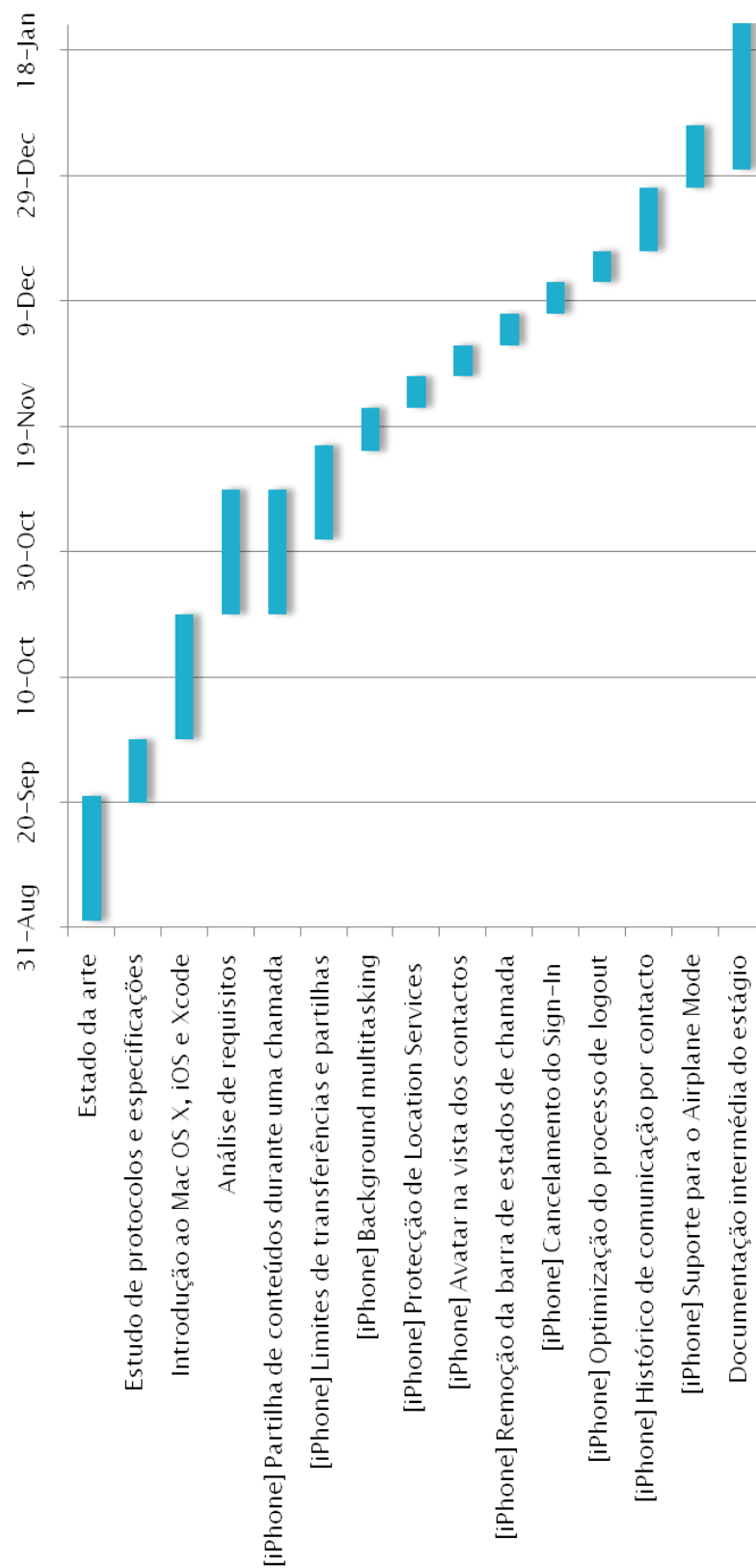


Figura 16 - Planeamento actual para o 1º semestre

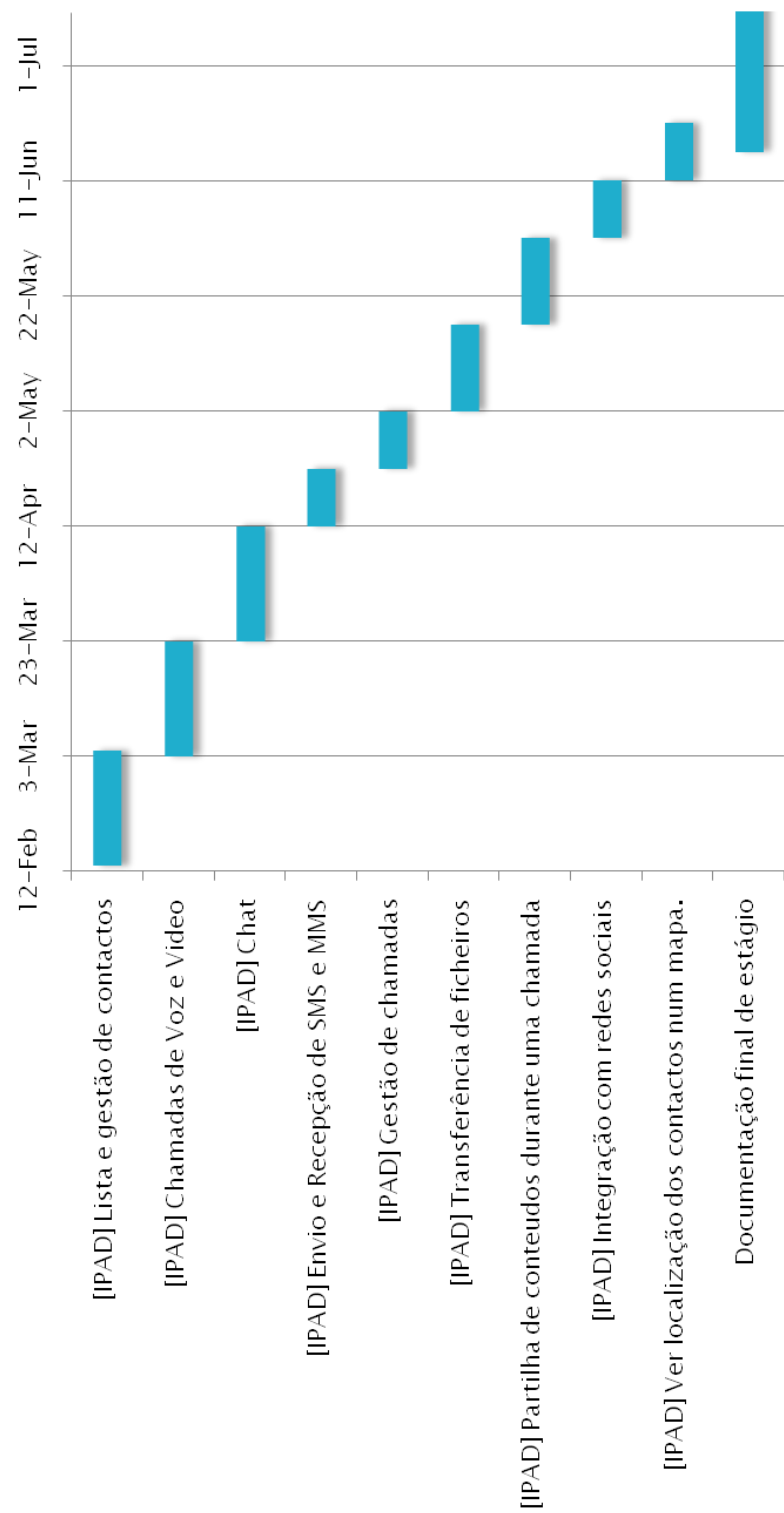


Figura 17 - Planeamento em Janeiro para o 2º semestre

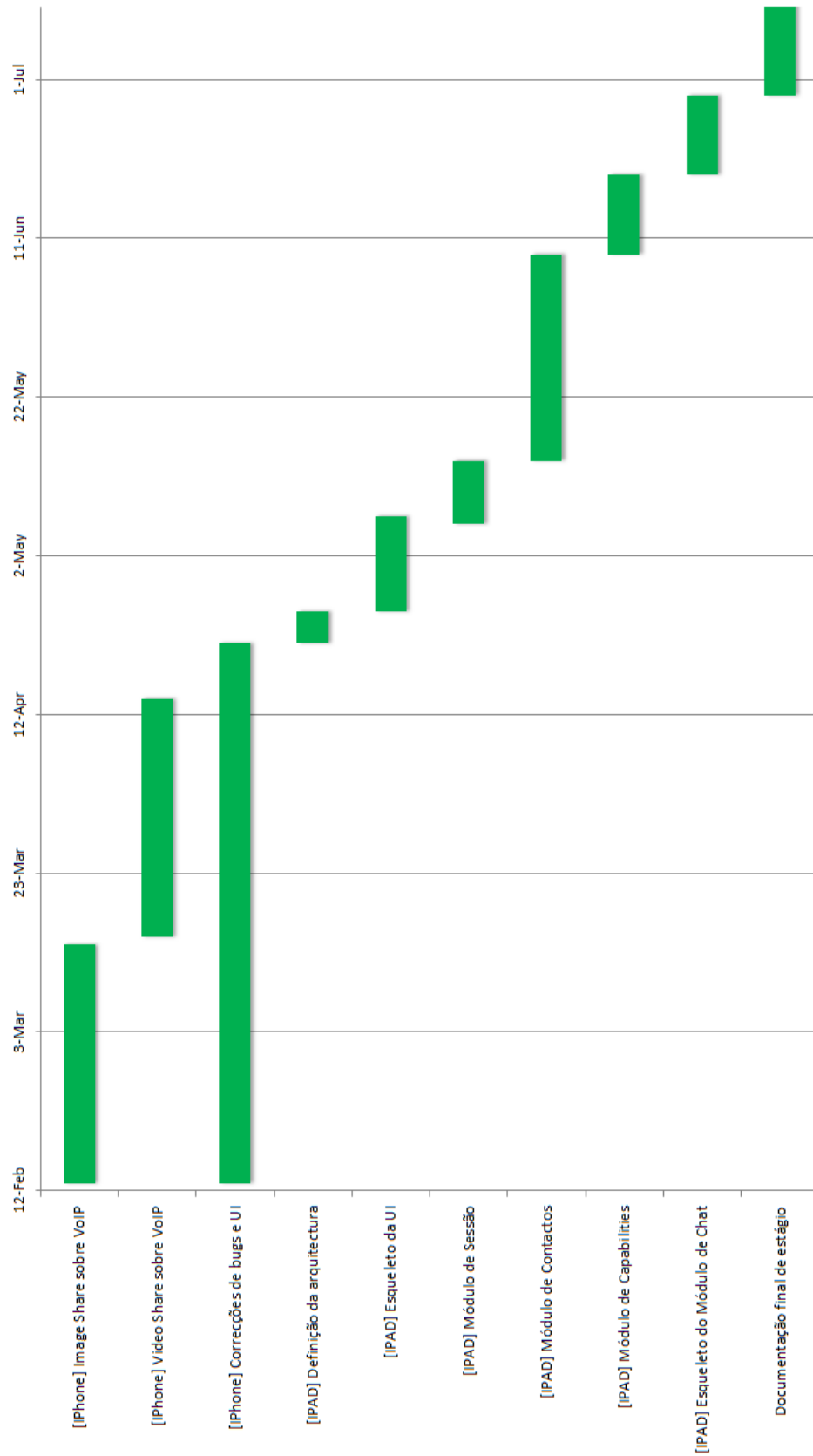


Figura 18 - Planeamento final para o 2º semestre

Capítulo 5

Resultados

De seguida apresentam-se alguns resultados do trabalho efectuado ao longo do estágio.

5.1 WMC iPhone – 1º Semestre

No primeiro semestre o estagiário desenvolveu algumas funcionalidades na aplicação para iPhone. Nas secções que se seguem podem ser vistos alguns screenshots dessas funcionalidades, e um breve comentário relativo aos obstáculos encontrados.

Image Share numa chamada nativa



Figura 19 - Screenshots do image share na chamada nativa no WMC iPhone 1/3

O utilizador começa por iniciar uma chamada nativa utilizando o dialer da aplicação, ou seleccionando a opção correspondente na lista de opções de comunicação do contacto. O pedido ao iOS para realizar uma chamada nativa para um dado número é trivial, mas a especificação RCSe declara que no momento em que se estabelece a chamada entre dois contactos, deve ser efectuada uma troca de mensagens SIP OPTIONS.

O utilizador que envia uma mensagem SIP OPTIONS está na prática a requisitar ao destinatário da mensagem quais os serviços de comunicação que ele tem disponível (capabilities). Ao enviar a mensagem, o utilizador anexa as suas próprias capabilities, e se a mensagem de resposta for um 200 OK, então no corpo da resposta estão as capabilities do contacto com quem estabelecemos a chamada.

O estagiário implementou os mecanismos para efectuar esta troca de mensagens ao iniciar uma chamada nativa, alertando o utilizador da possibilidade de partilhar conteúdo com o contacto no caso de ter recebido um 200 OK com a capability de Image Share. Para este efeito fez uso do centro de gestão de chamadas nativas do SDK do iOS.

É de realçar que este aviso apenas pode ser lançado ao utilizador que efectua a chamada, e não ao utilizador que a recebe. Isto deve-se a uma limitação do iOS que impede que qualquer aplicação (exceptuando as aplicações nativas) tenha acesso ao número de um contacto numa chamada recebida. No entanto, mais à frente voltaremos a este ponto, e explicaremos como contornámos este obstáculo.

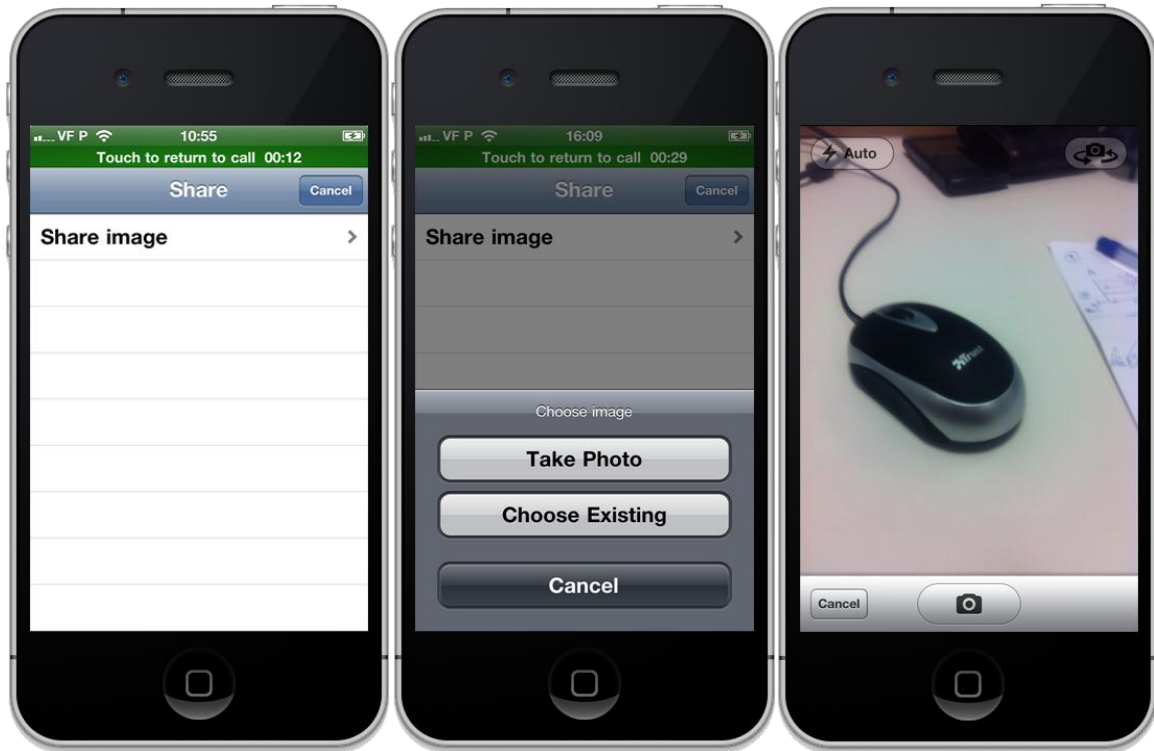


Figura 20 - Screenshots do image share na chamada nativa no WMC iPhone 2/3

No caso do utilizador pretender efectuar uma partilha de conteúdo, ele é redireccionado durante a chamada para a nossa aplicação. O estagiário implementou o mecanismo que detecta que o utilizador entrou na aplicação por motivos de partilha de conteúdo, o que permite então abrir um menu de partilha (embora neste momento tivesse apenas a opção de partilha de imagem, poderia ter também de partilha de vídeo ou de localização). Seleccionando a opção de partilha de imagem, o utilizador pode capturar uma fotografia ou escolher uma existente dos seus albuns.

Seleccionada a fotografia, é enviada uma mensagem SIP INVITE para o contacto. Essa mensagem leva consigo a descrição do ficheiro a enviar, o tamanho do ficheiro, e outras informações úteis para o contacto poder tomar uma decisão quanto à aceitação/rejeição do convite de partilha.

No caso do contacto aceitar o convite, o utilizador recebe uma mensagem 200 OK com um identificador da sessão MSRP estabelecida. A transferência é então iniciada por MSRP, e é mostrado ao utilizador o feedback de progresso/conclusão.

No caso do contacto rejeitar o convite, o utilizador recebe uma mensagem 603 Decline, permitindo que lhe seja mostrado feedback do motivo de insucesso da partilha (“Convite rejeitado”, por exemplo).



Figura 21 - Screenshots do image share na chamada nativa no WMC iPhone 3/3

Esta lógica foi implementada pelo estagiário, e os principais obstáculos encontrados consistiram nos casos específicos de cancelamento das transferências durante o seu progresso (conflito ou má gestão dos eventos lançados pela commlib), ou nas situações de partilha com dispositivos de outras plataformas. Um exemplo: ocorreram situações em que uma partilha era imediatamente terminada após o início quando era realizada numa chamada nativa com o Android. Estes casos eram complicados de resolver, pois nem sempre o facto da partilha funcionar com sucesso entre 2 dispositivos da mesma plataforma significava que fosse essa a plataforma com a implementação correcta.

Cancelamento do Sign In



Figura 22 - Screenshots do cancelamento do sign in no WMC iPhone

Embora esta funcionalidade pareça (e na prática seja) muito trivial, na verdade trouxe ao estagiário um conhecimento muito importante e útil para o restante tempo de contacto com a aplicação. Isto prendeu-se com o facto desta funcionalidade o ter posto pela primeira vez em contacto com todo o módulo de sessão da Communicator Library, que é provavelmente o módulo mais complexo da biblioteca. Também a implementação do botão lhe trouxe conhecimento a nível de desenho gráfico em iOS, uma vez que não foi utilizada uma imagem para esse efeito.

A nível de trocas de mensagens SIP; o cancelamento de um sign in envia sempre uma mensagem SIP REGISTER com um campo Expires: 0. Isto é na prática um un-register no servidor que até poderia não ser necessário (se não tivesse chegado a enviar o register inicial), mas que não causa qualquer entropia no servidor pois é descartado se não for esperado.

Edição do avatar de um contacto



Figura 23 - Screenshots da edição do avatar de um contacto no WMC iPhone

A edição do avatar de um contacto foi uma funcionalidade que permitiu ao estagiário trabalhar com o redimensionamento e corte de imagens no iOS e com o acesso aos álbuns de imagens do dispositivo. O redimensionamento foi algo de fulcral pois ao início a atribuição de uma imagem muito pesada a um contacto causava muitos problemas ao fazer scroll na lista de contactos, devido ao tempo de carregamento das células da tabela (o scroll “arrastava-se”). Outro obstáculo consistia nas situações que a imagem era alterada e posteriormente cancelada a alteração. Isso implicava que a edição do contacto fosse guardada temporariamente para ser confirmada apenas quando o utilizador premisse o botão de “Done”.

Histórico de comunicação de um contacto



Figura 24 - Screenshots do histórico de comunicação de um contacto no WMC iPhone

A implementação do histórico de comunicação permitiu ao estagiário entrar em contacto com componentes muito diferentes da aplicação. Por um lado, com as operações de armazenamento das mensagens na base de dados do utilizador no espaço da aplicação; por outro lado, uma vez que o histórico é visualizado nas salas de chat com os contactos, foi possível trabalhar com o Javascript e com o CSS no preenchimento da webview utilizada para o chat.

5.2 WMC iPhone – 2º Semestre

Nos primeiros meses do segundo semestre, o estagiário foi realocado para um projecto semelhante ao do primeiro semestre, ficando incumbido da responsabilidade de desenvolver duas funcionalidades: o image share durante uma chamada de VoIP, e o video share durante uma chamada de VoIP.

Image Share numa chamada VoIP



Figura 25 - Screenshots do Image Share em VoIP no WMC iPhone

O image share numa chamada VoIP funciona de forma muito semelhante ao image share numa chamada nativa (especialmente a nível de protocolo SIP). No entanto, uma vez que a chamada de VoIP pertence à aplicação, e tem uma vista que possibilita várias interações, foi um desafio interessante implementar esta feature sem interferir com os restantes casos de uso durante a chamada.

Para além disso, como esta aplicação disponibilizava também partilha de imagem em chamada nativa, teve de existir um cuidado especial a distinguir os casos de convites de partilha durante a chamada nativa dos convites durante a chamada VoIP.

Video Share numa chamada VoIP

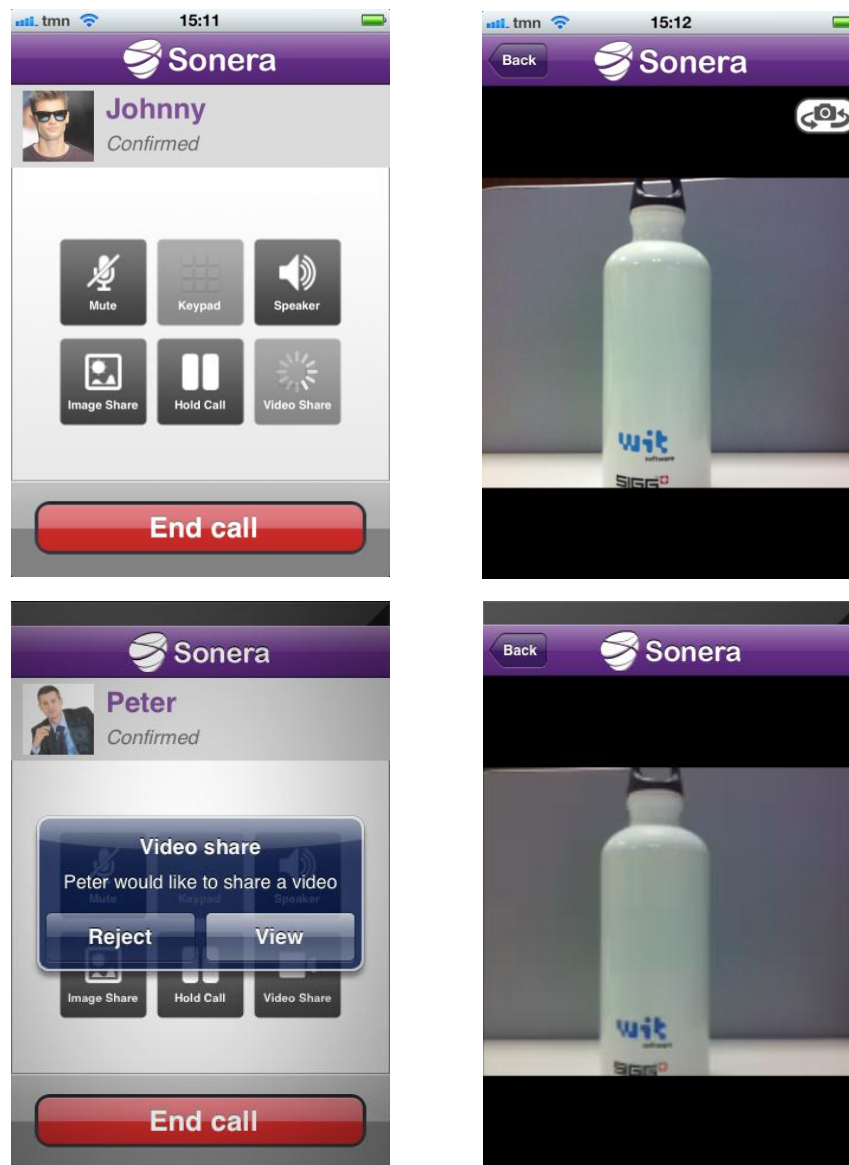


Figura 26 - Screenshots do Video Share em VoIP no WMC iPhone

O Video Share revelou-se ser uma funcionalidade mais problemática que o Image Share por vários motivos:

- A nível do dispositivo, o arranque da captura de vídeo é um processo relativamente lento. Isso criou a necessidade de dar algum feedback de espera ao utilizador (através de um spinner) e de bloquear a restante interface (para impedir, por exemplo, que o utilizador seleccionasse um image share imediatamente a seguir a seleccionar um video share). Um dos problemas mais complicados de resolver consistiu na situação em que um contacto desligava a chamada no instante em que o utilizador tinha

acabado de iniciar um Video Share. A solução passou por validar em vários pontos se a chamada continuava activa.

- O facto da Communicator Library ser utilizada por diferentes plataformas fez com que tivesse de lidar com hardwares muito diferentes. Algumas plataformas gravam o vídeo numa rotação diferente de outras, outras têm uma capacidade de processamento das frames muito lenta, etc. Para manter a portabilidade e o uso da biblioteca, tiveram que ser feitos alguns trade-offs, tais como separar o código na commlib para lidar da melhor forma possível consoante a plataforma, reduzir o número de frames enviadas por segundo e diminuir a resolução da captura.

A nível protocolar, o Video Share utiliza também a mensagem SIP INVITE para convidar o contacto para a partilha, mas seguidamente utiliza o protocolo RTP para o envio do vídeo.

5.3 WMC iPad

A meados do segundo semestre, foi retomado o plano inicial e foi dado o início do desenvolvimento de raiz da aplicação iPad. As tarefas realizadas consistiram em: definir a arquitectura da aplicação, criar o esqueleto de UI da aplicação, e implementação dos módulos de sessão, contactos, capabilities e chat.

Arquitectura da aplicação

O desenvolvimento de raiz da aplicação para o iPad foi uma excelente oportunidade de corrigir alguns erros cometidos na aplicação para o iPhone. Como as aplicações partilham a Communicator Library, as modificações que poderiam ser feitas seriam a nível da camada acima (lógica da aplicação no iOS e User Interface). Um dos maiores cuidados na definição da arquitectura foi o de isolar as diferentes camadas o máximo possível e definindo muito bem as suas funções (algo que não se sucedia na aplicação para iPhone). O diagrama de classes da página seguinte ilustra a arquitectura definida. Abaixo segue uma descrição simples do diagrama.

- Uma primeira análise ao diagrama permite perceber que se pretende modularizar ao máximo a aplicação, facilitando a adição, alteração ou remoção de módulos à medida que surjam novos requisitos ou sejam eliminados requisitos existentes.
- As linhas a tracejado representam eventos que são lançados e escutados por um dado módulo ou controlador. No caso das bridges, este é o ponto de entrada na parte da aplicação desenvolvida de raiz. Todas as mensagens recebidas pela aplicação de uma forma ou de outra despoletam um evento na CommLib que é capturado pela Bridge do contexto correspondente.
- As bridges então não servem (ou não se pretende que sirvam) para nada mais do que fazer uma tradução da informação e modelos que nos chegam da Communicator Library em C++ para informação e modelos em Objective C (e vice-versa nos pedidos realizados à CommLib).
- As linhas sólidas a preto simbolizam chamadas de métodos conhecidos. Depreende-se então que os Managers da camada de iOS são sempre informados dos eventos capturados pelas bridges, e são também eles os responsáveis por pedir às bridges que efectuem por sua vez os pedidos à CommLib.
- A camada de User Interface é composta por uma vista típica de iPad composta por duas vistas: uma vista Master onde tipicamente o utilizador selecciona uma opção de uma lista de opções, e uma vista de Detail que é construída/inicializada de acordo com a opção escolhida na vista Master.

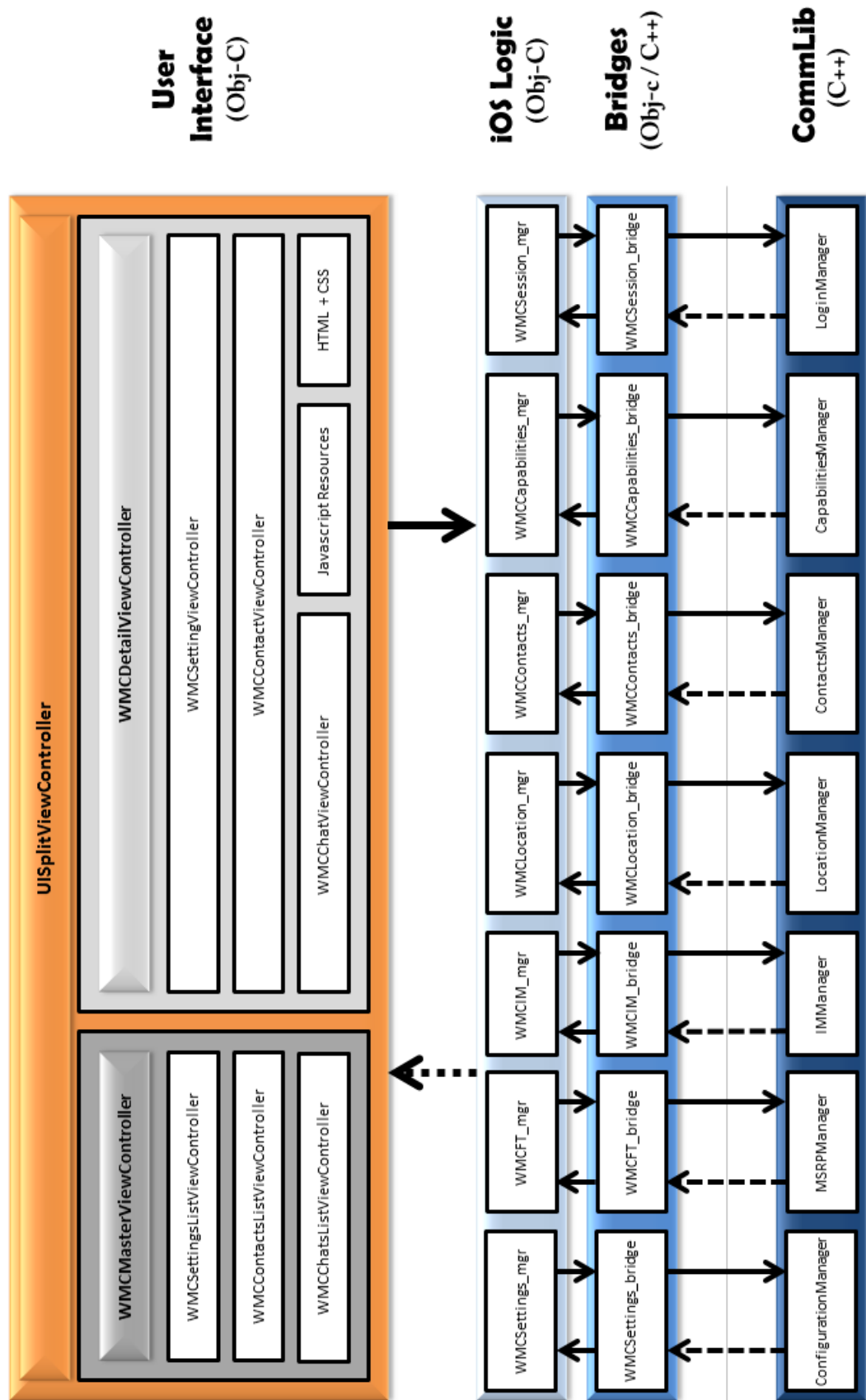


Figura 27 - Diagrama de classes da aplicação WMC iPad

Esqueleto da interface com o utilizador

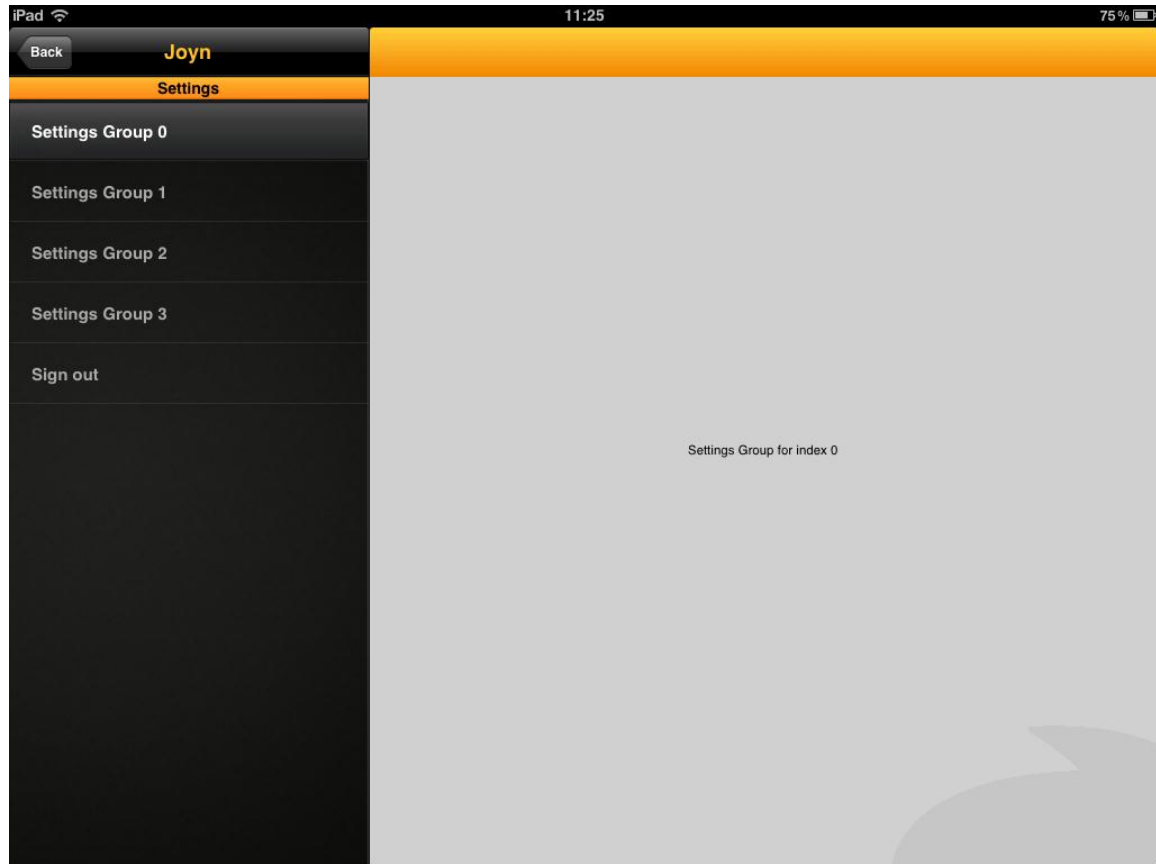


Figura 28 - Esqueleto da UI do WMC iPad

O esqueleto da interface consiste essencialmente na definição do controlador Master e do controlador Detail. O controlador Master terá sempre a um dado instante um controlador activo. Esse controlador é o responsável pela definição do controlador da vista de Detail. Para melhor entender o conceito, podemos ver na imagem acima que o `WCMMasterViewController` tem activo o `WMCSettingsListViewController`. Quando o utilizador selecciona a primeira opção (“Settings Group 0”), então o Master comunica ao `WMCDetailViewController` que deve instanciar um `WMCSettingViewController` (que neste caso simplesmente mostra uma label com o índice da opção).

Módulo de sessão

O módulo de sessão é o responsável pelo registo do utilizador no servidor e pela resposta às mudanças de conectividade da aplicação. Serve também como ponto central para os controladores na aplicação saberem em que modo de ligação a aplicação se encontra (online, offline, ou num estado de transição).

Para a detecção de conectividade a aplicação utiliza uma classe fornecida pela Apple de nome `Reachability`. Esta classe abstrai as implementações de mais baixo nível, e permite que uma aplicação que a utilize apenas registre um método como listener que será chamado de cada vez que se verificar uma mudança na conectividade. Para os efeitos de login, o módulo de sessão faz ainda as validações necessárias aos dados de registo do utilizador (estes devem estar inseridos nas definições da aplicação dentro das Definições Gerais do iPad).

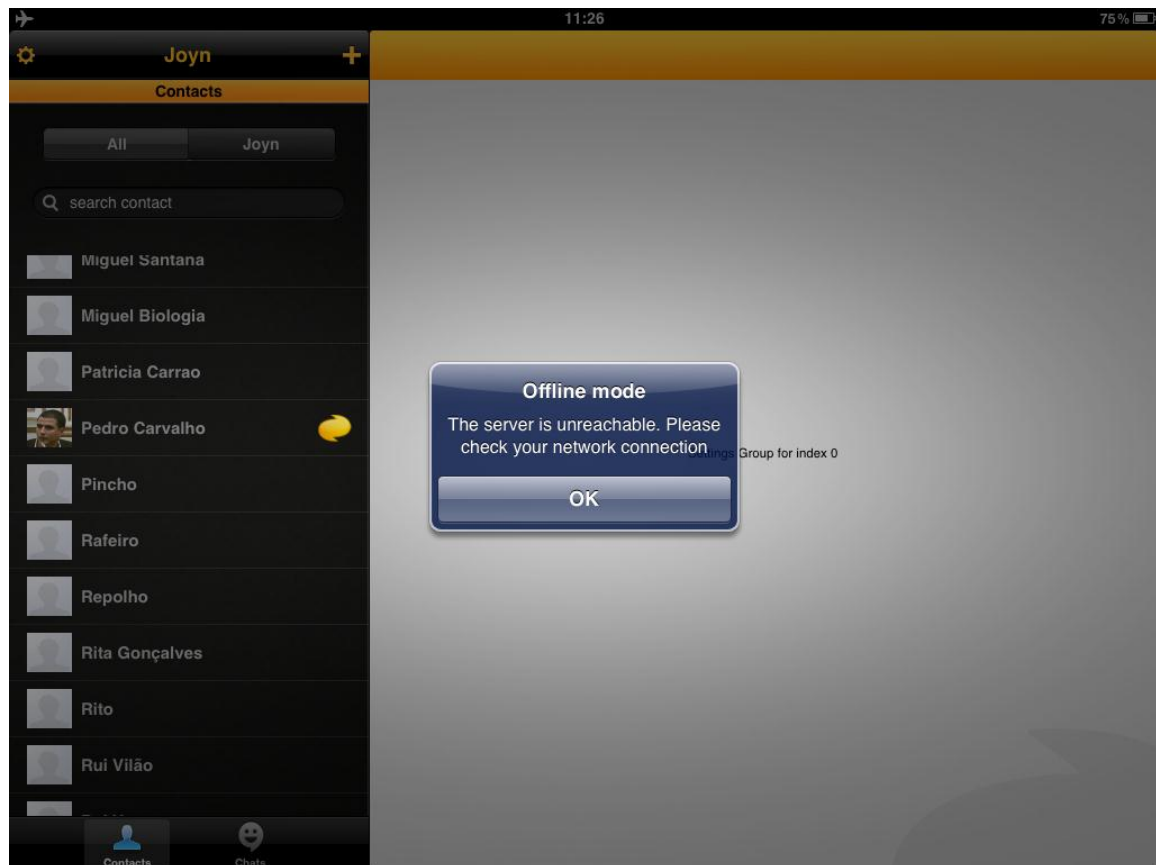


Figura 29 - Perda de ligação na aplicação WMC iPad

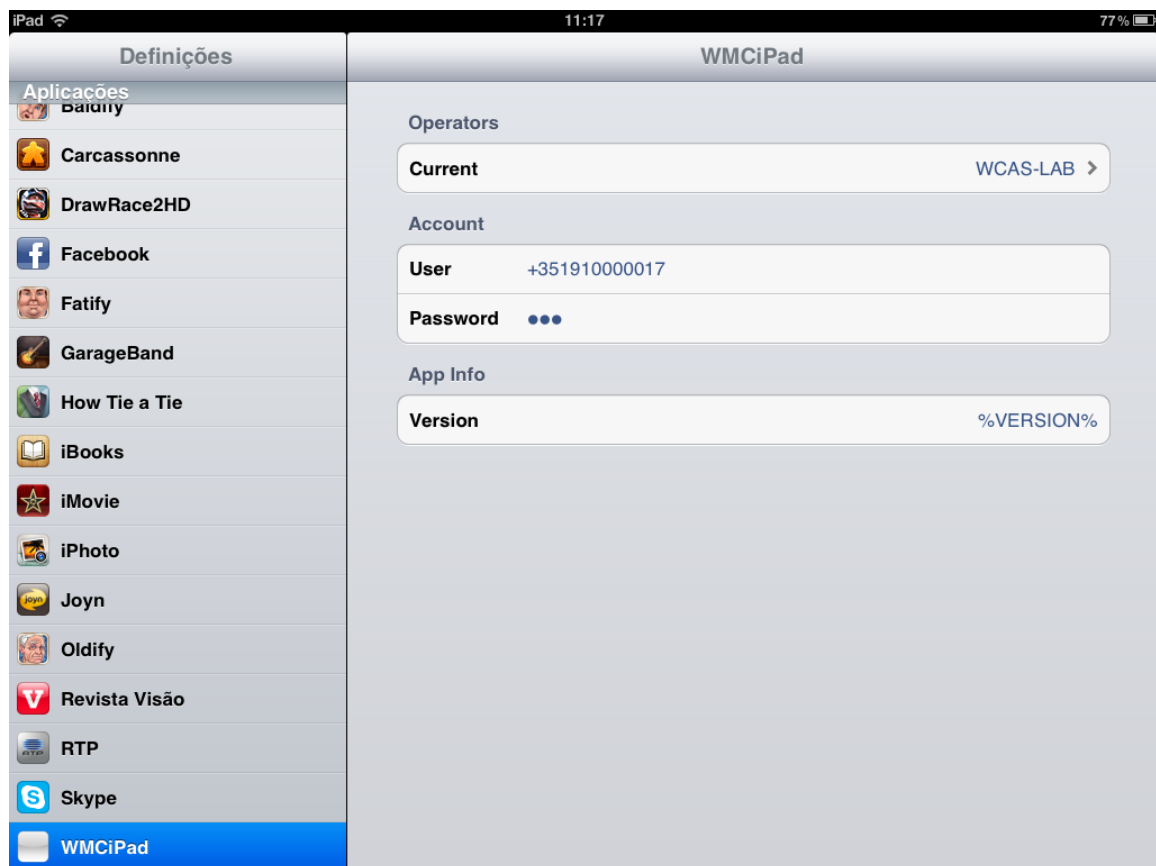


Figura 30 - Definições da aplicação nas Definições Gerais do iPad

Módulo de contactos

As funcionalidades deste módulo são facilmente previsíveis. A lógica da adição, edição e eliminação de contactos passa toda por aqui, assim como a actualização dos serviços disponíveis de cada contacto (numa interação por notificações com o módulo de capabilities). Sendo a lista de contactos da aplicação uma cópia da lista de contactos nativa, a sincronização desta lista com a lista nativa é também outra das funcionalidades deste módulo.

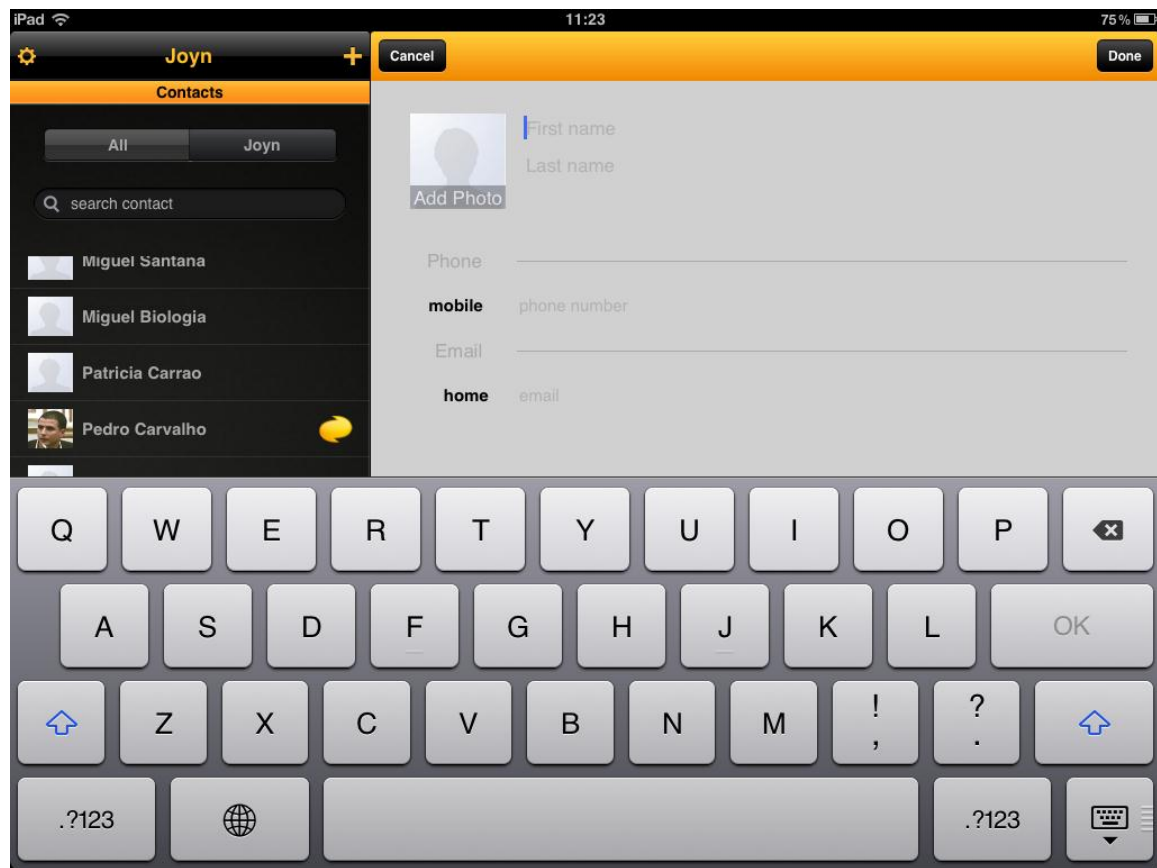


Figura 31 - Adição de um contacto novo no WMC iPad

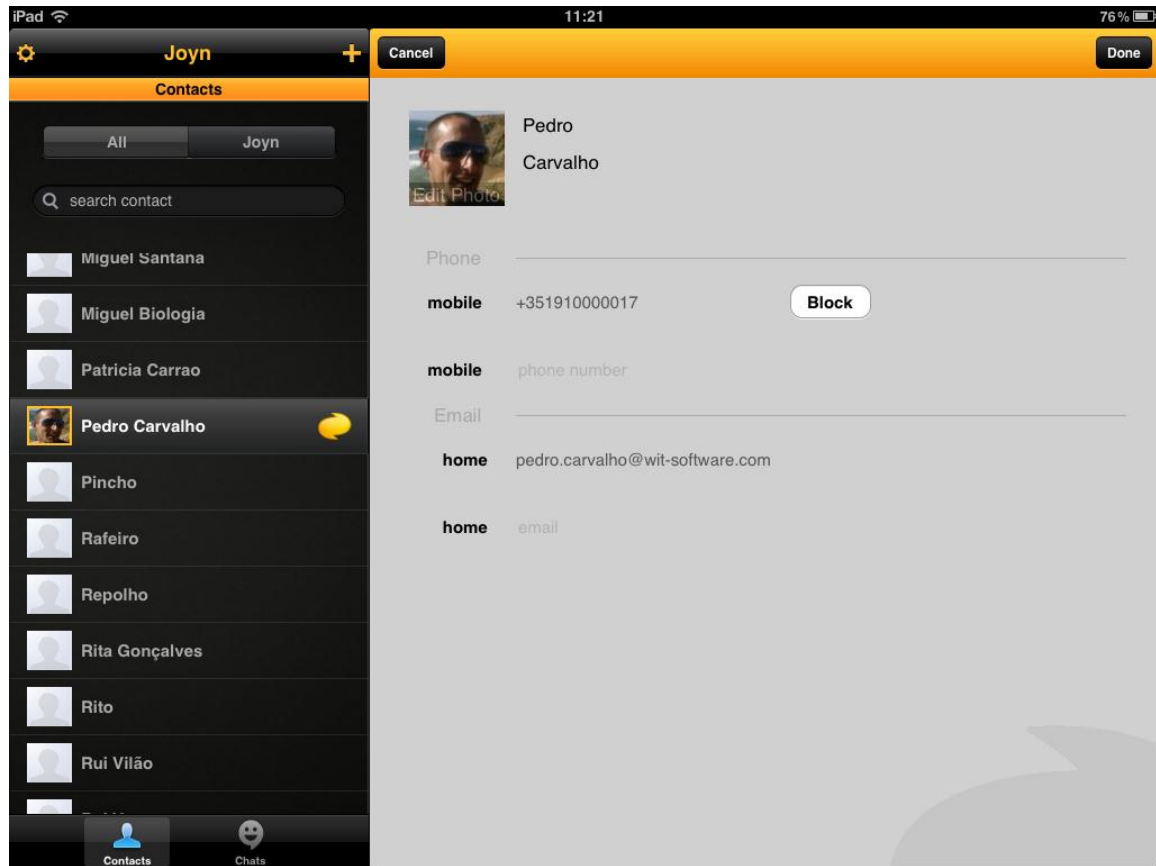


Figura 32 - Edição de um contacto no WMC iPad

Módulo de capabilities

Na imagem acima é possível verificar que o contacto “Pedro Carvalho” tem serviços Joyn/RCS disponíveis (através do ícone amarelo sabemos que a última actualização de capabilities do contacto nos informou que o contacto está disponível para comunicações Joyn). Este tipo de informações resulta do trabalho do módulo de capabilities. É ele o responsável por avisar os seus observers da chegada do resultado de um pedido de capabilities para um dado contacto, ou simplesmente da chegada de um pedido das nossas capabilities. Os pedidos iniciados por nós também têm passagem obrigatória neste módulo.

Módulo de chat

No momento do término do estágio apenas estava desenvolvida uma pequena parte da funcionalidade do chat. No entanto, o estagiário implementou o esqueleto da bridge de Instant Messaging, o esqueleto do manager de Instant Messaging, e os métodos de Javascript que conjuntamente com as definições de CSS permitem utilizar uma webview como uma sala de chat.

Com algumas chamadas de funções hard-coded, foi possível testar o comportamento esperado quando o módulo estiver funcional (visível na imagem seguinte).

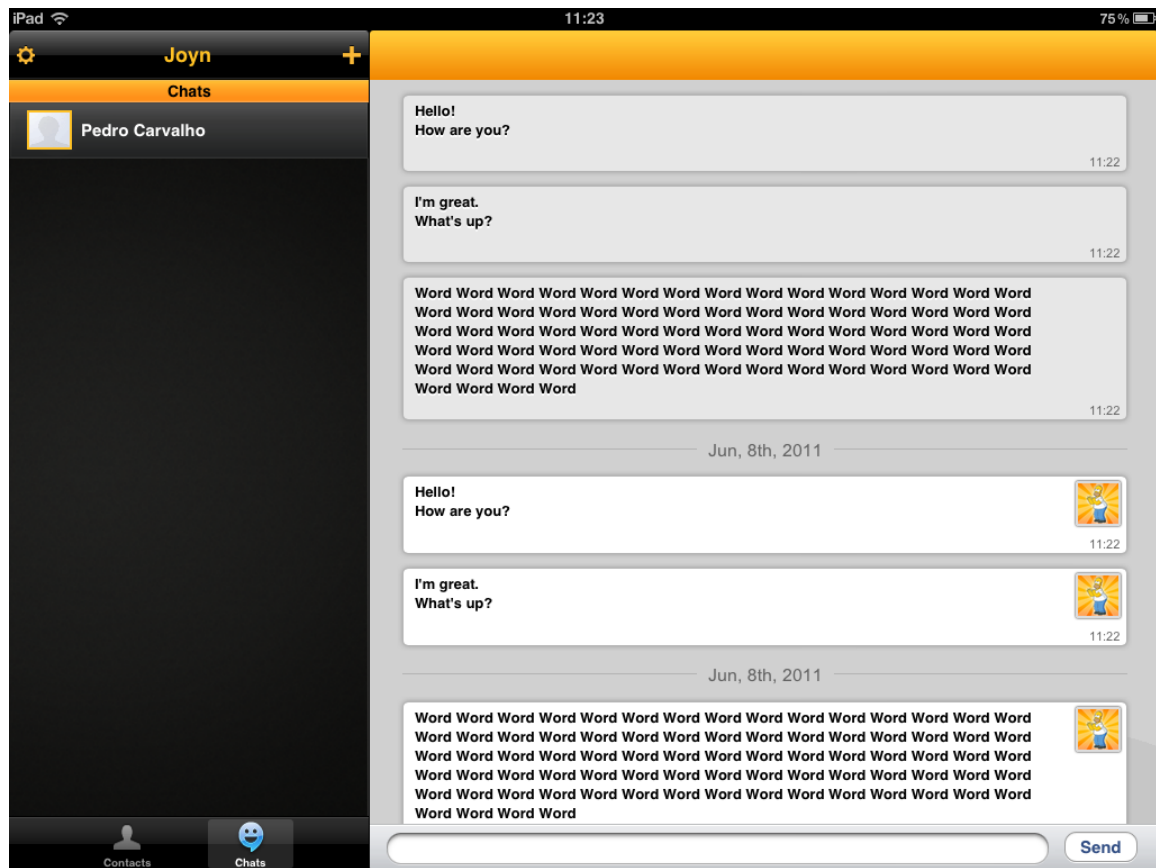


Figura 33 - Vista de chat no WMC iPad

Capítulo 6

Conclusões

Este estágio teve a particularidade de estar associado a um projecto de elevada ambição e responsabilidade, sendo neste momento sem dúvida o projecto mais importante da WIT Software. A empresa encontra-se na linha da frente de um projecto global que ameaça revolucionar o mundo das telecomunicações tal como o conhecemos, tendo já adquirido uma visibilidade mundial graças à sua aposta no Joyn.

Em projectos desta dimensão o tempo de lançamento é um factor crítico de sucesso, e as oportunidades são tão escassas quanto efémeras. O elevado poder dos intervenientes (operadores móveis e GSMA) aumenta o risco do investimento a um ponto de impedir que as especificações estabilizem e convirjam para um standard único. O RCS e ainda não veio cá para fora “em força” e a GSMA já prepara o RCS5 para o vir substituir.

Uma das principais razões para este receio de falhar deve-se ao facto da procura ser grande. Os fabricantes dos equipamentos aguardam impacientemente as directrizes das especificações. Um lançamento prematuro simboliza uma perda muito grande de investimento, e um lançamento tardio é um atraso enorme na corrida para as comunicações da nova geração, o que pode simbolizar uma perda ainda maior. Os operadores móveis querem ser os primeiros do seu país a lançar os novos serviços, mas se não for um lançamento seguro, arriscam-se a perder imenso dinheiro. Num ponto todos têm a certeza: existe um potencial enorme nesta proposta de enriquecimento da comunicação.

Tendo sido esta a realidade em que o meu estágio se inseriu, consigo olhar para trás e compreender as mudanças de rumo que o planeamento sofreu. Certamente perdi a oportunidade de aprender muito com o desenvolvimento de uma aplicação única a longo prazo, e possivelmente prejudiquei-me a nível da documentação, mas apesar de tudo penso que não saí a perder.

Aprendi novas tecnologias e usei ferramentas que nunca tinha usado. Mas mais do que isso aprendi a trabalhar no seio de uma empresa, quer de forma colaborativa quer a desenvencilhar-me sozinho.

Ao fim de um ano de estágio na WIT Software posso afirmar que vivi uma experiência extremamente enriquecedora. Foi o meu primeiro contacto com o mercado de trabalho, e tive a oportunidade e felicidade de trabalhar com pessoas fantásticas e extremamente competentes com quem aprendi muito.

Referências

- [1] International Data Corporation Press Release: Framingham, Massachusetts; **Worldwide Smartphone Market Expected to Grow 55% in 2011 and Approach Shipments of One Billion in 2015**; <http://www.idc.com/getdoc.jsp?containerId=prUS22871611>
- [2] International Data Corporation Press Release: London; **Western European Consumers Moving on From Laptops to iPads?**; <http://www.idc.com/getdoc.jsp?containerId=prUK22863711>
- [3] International Data Corporation Press Release: Singapore; **Anytime, Anywhere Access to Social Networks and Instant Messaging Driving Demand for Increased Levels of Mobile Connectivity**; <http://www.idc.com/getdoc.jsp?containerId=prSG22907411>
- [4] Página oficial da GSM Association; <http://www.gsma.com/about-us/>
- [5] Página oficial do projecto Rich Communications Ecosystem; <http://www.gsma.com/project-ecosystem/>
- [6] Página oficial da Wit-Software: <http://www.wit-software.com/>
- [7] International Data Corporation Press Release: Framingham, Massachusetts; **IDC Forecasts Worldwide Mobile Applications Revenues to Experience More Than 60% Compound Annual Growth Through 2014**; <http://www.idc.com/about/viewpressrelease.jsp?containerId=prUS22617910>
- [8] Strategy Analytics Report Abstract Viewer: Patel, Nitesh; **Global Mobile Messaging Forecast (2001-2016)**; <http://www.strategyanalytics.com/default.aspx?mod=reportabstractviewer&a0=6535>
- [9] NPD Group Press Release: New York; **Consumers Now Take More Than a Quarter Of all Photos and Videos on Smartphones**; https://www.npd.com/wps/portal/npd/us/news/pressreleases/pr_111222
- [10] GSMA RCS-e specifications website: <http://www.gsma.com/product-specifications-rs-e-specifications/>
- [11] GSMA commercialization website: <http://www.gsma.com/commercialisation/>
- [12] Wit Communications Suite booklet: <http://www.wit-software.com/docs/wcs-booklet.pdf>
- [13] SIP Session Initiation Protocol RFC: <http://www.ietf.org/rfc/rfc3261.txt>
- [14] MSRP Message Session Relay Protocol RFC: <http://tools.ietf.org/html/rfc4975>
- [15] IMS IP Multimedia Subsystem: <http://www.ibm.com/developerworks/webservices/library/ws-soa-ipmultisub1/>
- [16] Osip: <http://www.gnu.org/software/osip/>
- [17] OpenSipStack: <http://www.opensipstack.org/>
- [18] reSIProcate: http://www.resiprocate.org/Main_Page
- [19] Metodologia Waterfall: <http://www.techrepublic.com/article/understanding-the-pros-and-cons-of-the-waterfall-model-of-software-development/6118423>

- [20] Metodologia Agile: <http://www.agile-process.org/>
- [21] Apple developer tools: <https://developer.apple.com/technologies/tools/>
- [22] Wireshark: <http://www.wireshark.org/>
- [23] svnX: <http://www.macupdate.com/app/mac/15527/svnx>
- [24] Mantis: <http://www.mantisbt.org/>