

# **Utilization of cloud computing in software products designed for children with autism**

by  
**Robert Peter**

MSc in Software Development  
Cork Institute of Technology

Supervisor: **Dr. Jeanne Stynes**

This report is submitted in partial fulfilment of the requirements for the Degree of Master of Science in Software Development at Cork Institute of Technology. It represents substantially the result of my own work except where explicitly indicated in the text.

XX  
XX

July 2011

# **Utilization of cloud computing in software products designed for children with autism**

by  
**Robert Peter**

## **Abstract**

There are a quite a few special tools and methods that are widely used when dealing with children who are affected by autism. A significant number of them are based on visualization – trying to provide a visual aid for those everyday tasks that are easily adopted by an average child but sources of difficulties for a child on the autistic spectrum. This project develops a framework which utilizes the features of cloud computing (particularly Google’s cloud) to provide visualization and cooperation when supporting children in the autistic spectrum in a cost-effective and intuitive way. An application prototype that uses this framework is also constructed.

## Table of Contents

1	Introduction.....	1
2	Background Research.....	2
2.1	Autism Spectrum Disorder.....	2
2.2	Managing Autism.....	3
2.3	Assistive Techniques and Systems.....	3
2.4	Assistive Techniques and Computers.....	4
2.5	Available Products.....	4
2.6	Survey on Software Products and User Experience.....	7
2.6.1	Survey aim and method.....	7
2.6.2	Survey evaluation.....	8
3	Design Proposal.....	11
3.1	Orientation.....	11
3.2	Why Cloud Computing?.....	12
3.3	Evaluating Google Cloud.....	13
3.4	Requirement Summary.....	14
3.5	Licence information.....	15
4	Infrastructure Design.....	15
4.1	Environment.....	15
4.2	System Components.....	16
5	Application.....	18
5.1	Design.....	19
5.1.1	User Stories.....	19
5.1.2	General Considerations.....	21
5.1.3	Pictogram translation.....	22
5.2	Implementation.....	24
5.2.1	Data Objects.....	24
5.2.2	Data Access Objects.....	26
5.2.3	PictoManager.....	29
5.2.4	Servlet Classes.....	32
5.2.5	Front-End Pages.....	35
5.2.6	System Overview.....	39
5.3	Testing.....	40
5.4	Release.....	42
5.5	Future enhancements.....	42
6	Conclusion.....	43
	References.....	45
	Appendices.....	47

# 1 Introduction

In this software development research project the goal is to provide a software infrastructure solution that enables different people from different locations to work closely together as a team sharing their efforts in helping a child who is on the autistic spectrum. In order to be able to create a tool that is really beneficial for those involved, a background research is carried out first to gain more knowledge in the problem domain. This background research includes a short summary about the autistic disorder and the assistive techniques that are used in conjunction with it. A quick look is taken into the currently available software products. A survey was performed and evaluated about user experience and future requirements. Then, based on the conclusion drawn from background research a design model is proposed, an infrastructure framework that could facilitate software applications to better fulfil the special requirements. Furthermore, an application prototype is developed that uses this model and is useful for the community. The model and the prototype utilize cloud computing to enable teamwork over Internet. The framework will facilitate creating intuitive graphical front-end applications and connect them to a general-purpose cloud architecture as back-end.

In one sentence, the research question can be expressed as the following: What are the utilization possibilities of cloud computing when creating software products aimed for children with autism and the team that looks after them? How can cloud computing be utilized for such special purposes?

The deliverable results of this project are:

- An **analysis of user requirements** in the domain, based on literature review and a survey
- An **infrastructure framework** that utilizes cloud computing to enable teamwork and provides visualization solutions to use in several different applications.
- An **application prototype** – a visual schedule, based on Google Calendar – that fits in that framework and can be used by the people who work with supporting an autistic child.

## 2 Background Research

### 2.1 Autism Spectrum Disorder

Autism and related disorders are known to the professionals in the area since the 1940's. Such mental condition, however, has references in the literature from the last couple of centuries. As classified in **DSM-IV-TR** (*Diagnostic and Statistical Manual of Mental Disorders, 4<sup>th</sup> Edition, Text Revision*, the most recent diagnostic manual to date)<sup>1</sup>, it belongs to the Pervasive Developmental Disorders. There are 5 disorders listed in this group, namely: Autistic Disorder, Asperger's Syndrome, Childhood Disintegration Disorder, Rett's Syndrome, and Pervasive Developmental Disorder, Not Otherwise Specified (PDD-NOS). Three out of these five are usually referred as Autism Spectrum Disorders (ASD), they are: Autistic Disorder, Asperger's Syndrome and PDD-NOS. The term 'autism' usually refers to ASD rather than only Autistic Disorder.

ASD is typically diagnosed between 3 and 5 years of age, depending on the severity; it affects each individual to different degrees. The impairments are from 3 core domain:

- communication and language
- socialization abilities
- restricted interests (which later results in imbalanced knowledge) and repetitive, unimaginative and stereotyped patterns of behaviour and play.

Deficits in this triad distinguish autism from other developmental disorders. Autistic children often lack the intuition about others and have difficulties interpreting gestures or facial expressions. Making and maintaining friendships is hard for those with autism. Up to a half of children with autistic disorder do not develop speech to a level that would be required for their daily communication needs.<sup>2</sup>

---

1 APPI, 2000. *Diagnostic and Statistical Manual of Mental Disorders, 4th Edition, Text Revision (DSM-IV-TR)* 4th ed., United States: AMERICAN PSYCHIATRIC PRESS INC (DC).

2 Quinn, B., 2011. *Autism spectrum disorders□: an altered perspective* 2nd ed., London: Jessica Kingsley Publishers.

Unfortunately, it is still not known by the scientists, what root causes are responsible for ASD. As reported in a recent journal article<sup>3</sup>, prevalence of ASD is surprisingly high. While earlier it was estimated to fall between 5 and 20 per 10,000 – the number of diagnosed cases shows a rapid increase since the 1990's. Studies from the 21<sup>st</sup> century report 60-110 cases of ASD in 10,000 children. Investigation about the reason of such a rapid increase is ongoing and while there are several possible causes considered, as of now there is no wide agreement on any of those hypotheses.

## **2.2 Managing Autism**

There is no medication known that would treat the core symptoms of Autism Spectrum Disorders. While medications are prescribed to address certain behavioural problems, ASD as a whole is best treated using educational interventions. These interventions address the above mentioned three groups of impairments. Disabilities in communication and language skills are targeted by speech and language therapy. Social skills are developed by occupational therapy, role play, social skill trainings or social games. Deficit in interest and concentration needs to be dealt with in the school: resource teachers and special need assistants help the work of the class teacher.

## **2.3 Assistive Techniques and Systems**

There are various methods and techniques used both during the therapies and in the everyday life of the children. Several methods are based on pictures, drawings or some other kind of visualisation. Communication with those children that did not develop the ability of speech happens using pictures. PECS (Picture Exchange Communication System)<sup>4</sup>, for example, is a widely approved approach that uses pictures to develop communication skills. It is found to be appropriate for children with a range of learning, speech and communication difficulties, including autism. Social stories are usually also supported by drawings describing the situation. When

---

3 Newschaffer, C.J. et al., 2007. The epidemiology of autism spectrum disorders. *Annual Review of Public Health*, 28, pp.235-258.

4 Frost, L. & Bondy, A.S., 1999. *PECS training manual*: the Picture Exchange Communication System [1999 ed.], Cherry Hill NJ: Pyramid Educational Consultants.

teaching something or trying to introduce a daily routine, visual illustration is again an important aid that is often used.

## ***2.4 Assistive Techniques and Computers***

Visualisation, visual aids, picture exchange communication. Even more than in the mainstream education, using different visual clues has a key importance for professionals and parents who deal with an autistic child. Among others, visualisation is an important area where computers can provide assistance to those who work with children on the autistic spectrum. There is a wide variety of educational and support applications available. It is not the aim of this report to provide a detailed review of the currently available digital tools but a certain insight is necessary to learn about strengths and shortcomings of available products.

## ***2.5 Available Products***

Presently, there are numerous applications available for several different purposes, all aiming at helping parents, teachers, therapists. Presumably the majority of them provide useful functions and help people a lot. Still, there seems to exist a gap that would be of interest for later development. When looking around for assistive applications, the ones we can find usually fall under one of the following categories:

- Standalone applications
- Client-server application systems
- Complete web applications

Programs in the above three categories all have their advantages and disadvantages. They all have their well defined use and there are no doubts that they are of great help. Standalone applications can be used at home or at the school, by the parents or by the teacher/therapist. They rarely support co-operation. More and more popular are the applications that are developed for mobile devices, using the Android or iPhone/iPad platform. Just to mention one out of hundreds, Grace App<sup>5</sup> is a development from Ireland, which is a communication support for people with

---

5 Domican, L, Grace App for Autism on iPhone, iPad and iTouch. Available at: <http://graceappforautismoniphone.blogspot.com/> [Accessed June 8, 2011].

autism: it is using pictograms to form sentences so the autistic person can communicate without talking.

Client-server application systems were also developed as support for institutes dealing with autistic children. A very good example is vSked, which is a visual scheduling and demonstration tool. It provides a magnificent assistance, primarily for schools that deal with several kids. It was introduced in 2010 at two different conferences<sup>6 7</sup>. Because of the architectural needs that are implied in such products (needs server, clients, network infrastructure, administration tasks), they are aimed at special institutes. It is not likely that a mainstream school that only a few autistic children attend could afford such resource-intensive systems. Another example is the HANDS project, which is still in research/development phase but its architecture presumes the existence of a serious (and expensive) background infrastructure that ensures the functionality of the hand-held device of the end users<sup>8</sup>.

Complete web sites / web applications were also created to support children with autism. While they usually provide resource sharing and are available for individuals as well as organizations, they are usually created uniquely to aim at a certain problem area. Development of such sites is resource-intensive and it also needs continuous administration, support. Further expenses like hosting and maintenance seem unavoidable for such an application system. As an example, a very interesting initiative is ChARM Tracker<sup>9</sup>, where parents, professionals, doctors can track treatment details, observations. A real co-operation tool which is also a great resource for researchers.

---

6 Hirano, S.H. et al., 2010. vSked: evaluation of a system to support classroom activities for children with autism. In *Proceedings of the 28th international conference on Human factors in computing systems*. Atlanta, Georgia, USA: ACM, pp. 1633-1642.

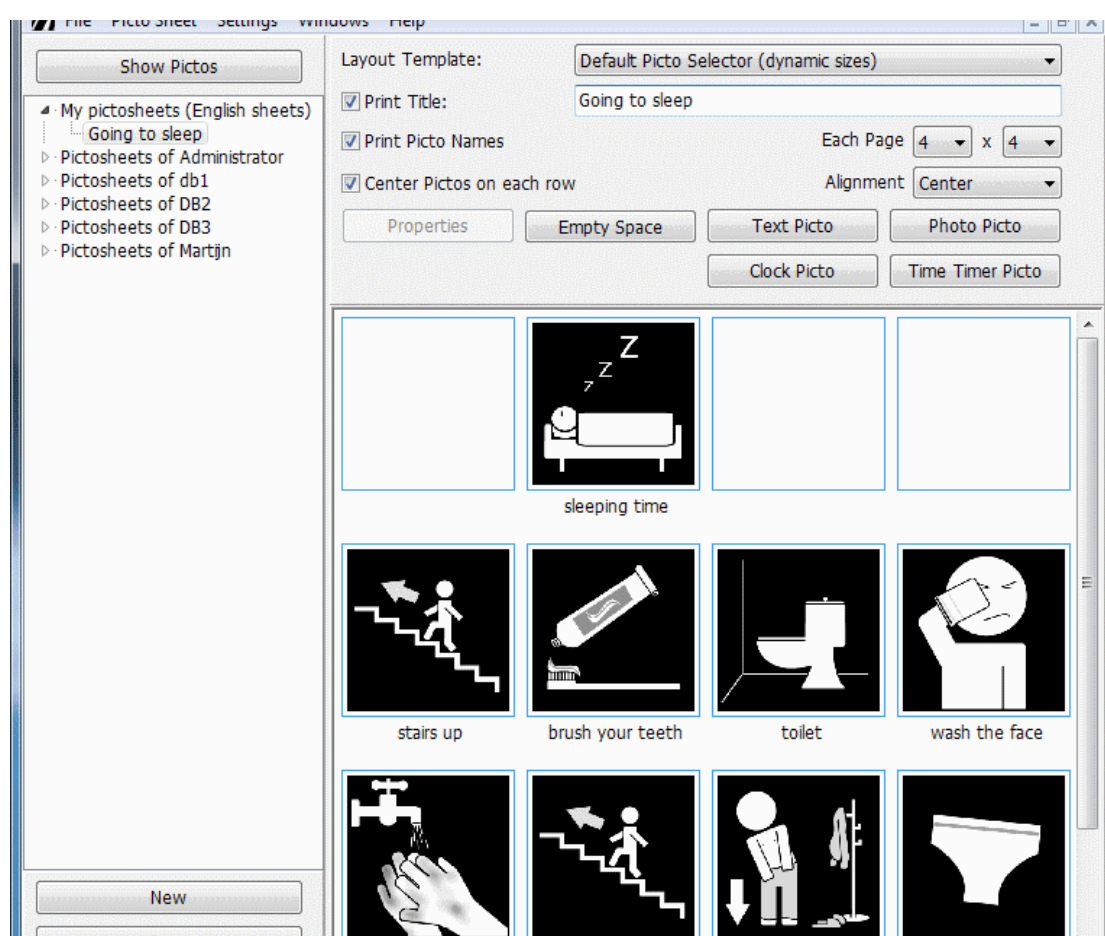
7 Yeganyan, M.T. et al., 2010. vSked: an interactive visual schedule system for use in classrooms for children with autism. In *Proceedings of the 9th International Conference on Interaction Design and Children*. Barcelona, Spain: ACM, pp. 319-322.

8 Ranfelt, A.M., Wigram, T. & Øhrstrøm, P., 2009. Towards a handy interactive persuasive diary for teenagers with a diagnosis of autism. In *Proceedings of the 4th International Conference on Persuasive Technology*. Claremont, California: ACM, pp. 1-9.

9 MedicalMine Inc, ChARM Tracker. *Cloud-based affordable healthcare solutions*. Available at: <https://charmtracker.com> [Accessed June 11, 2011].



There is an application that gave direct inspiration to come up with the idea of utilizing the cloud for visualization-related software products. **Picto Selector**<sup>10</sup> is a freely available application which is a great help in organizing pictograms, creating instruction sheets, visual schedule. This application is built around the idea of putting together sheets built of pictograms, aimed to tell a social story or instructions for a child. It has a remarkable amount of pictograms and provides tools to create further ones. It makes use of a freely available pictogram set that is published under the Creative Commons licence and contains tens of thousands little images. The same set is used by the application framework in this project.



*Drawing 1: Screen shot of Picto Selector*

<sup>10</sup> van der Kooij, Martijn, PECS for All. Available at: <http://pecsforall.com/> [Accessed April 27, 2011].

## **2.6 Survey on Software Products and User Experience**

### **2.6.1 Survey aim and method**

In order to get a picture of the current situation about the available software products and their usage, a survey was conducted and evaluated. The aim of the survey was to gain information from those who are potential users of assistive computer applications. While there seem to be several different software products available, one can have the impression that they are not widely used. Based on a similar survey conducted in the United States a few years ago<sup>11</sup>, demand and offer do not meet: *“The results indicate that excitement expressed in the literature about a seemingly good fit between technology and people with ASD does not translate to a high proliferation of products in user’s homes; only 25% of responders had any experience with software and technology designed for people with cognitive disabilities and only eight of the 45 reviews were about products specifically designed for people with ASD”*.

I wanted to gather representative data about the current situation in Ireland: are there software products in use to help children with ASD? Is there a need for such products? Are there any concerns with using these applications?

The survey was performed online, using the *KwikSurveys* web site. The link to the survey was distributed in schools, voluntary organizations and among professionals that deal with children on the autistic spectrum. Distribution was targeted, the link – while it was available on the public Internet, without restriction – was not advertised in any uncontrolled ways, distribution was targeted to those that are indeed exposed to autism. Only adults (above 18 years of age) were allowed to respond. There were 86 people approached altogether but they were encouraged to pass the link on to anybody who satisfies this criteria (i.e. has interactions with autistic children) so the total count of candidates could have been around 100. Nobody was asked afterwards if they filled in or passed it on to others. Participation was voluntary and completely anonymous. There was no supervision at all, everybody filled in the survey on their

---

11 Putnam, C. & Chong, L., 2008. Software and technologies designed for people with autism: what do users want? In *Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*. Halifax, Nova Scotia, Canada: ACM, pp. 3-10.

own discretion and individually. Generally speaking, the answers do not unveil the person who gave them. A printed version of the survey can be found in Appendix #1.

### **2.6.2 Survey evaluation**

The detailed evaluation of the survey can be found in Appendix #2. Below is a summary of the relevant findings.

**(A) Participants:** The final number of the completed answers is 25. 16 of them were people who are in contact with several autistic children, 9 have contact with one particular child only. The majority of the respondents were parents (10), then class teachers (5), resource teachers (3), special need assistants (3), therapists (3) and 1 family member other than parent. In other words, slightly less than half of the people (11) who gave answers were from the family while the other half (14) were those professionals that work with autistic children. So we have answers from both major perspectives.

**(B) User Experience:** the majority of the respondents used some kind of supporting/ educational software product. Only 3 used none, 14 selected one category, 4 selected two, 2 selected three, 1 selected four and 1 selected six (who is actually a therapist, dealing with several children). The most popular category was the software products designed for children (13), then the ones for generic use (7) – applications aimed specifically for children/people with cognitive disabilities or ASD were in minority. Presuming that a significant part of people who did not fill in the survey probably did so because they do not use software products, still at least 22 out of approximately 100 people did try to help their work with an autistic child using some kind of software product.<sup>12</sup> Answers on another question showed that cooperation over network was either not offered by the products or could not be utilized in more than half of the cases. The average satisfaction index was 3.55 (on a scale 1 to 5). In addition to that, 20 participants indicated using general purpose software products in relation to supporting a child and 23 reported using resources from the Internet.

When listing actual software products that they were using, the two most used categories are the educational and the visualization-based pieces of software,

---

<sup>12</sup> This result is very much in accordance with the results from Putnam and Chong (2008), as they reported that 25% of responders had any experience with software.

followed by generic office applications and products for social development. Most of the educational pieces of software do utilize some kind of visualization as well. We can rightfully presume that office applications were also mostly used to make visual aids. Also the social development applications rely on social stories, perceived visually.

Purpose of use was also asked, the responses tell us that (apart from entertainment and games) the most popular reasons are: *learning aid, language development, visualization, activity schedule, communication skill development and social skill development.*

Finally, about the hardware in use: The most used piece of hardware is still the personal computer (22). While there are special assistive product available, they are in rare use (2) in homes or smaller organizations, probably because of the higher price. Mobile devices (1) are not very popular, reason is probably the limited size of the screen. Tablets are getting more and more popular in general and as it seems, the number of devices used (5) confirms the market trend.

**(C) Needs and Concerns:** One question asked about the autistic children's attitudes and behaviour toward computer technology: The majority of respondents confirmed the widely sounded belief about the good fit between computer technology and children with autism. Some concerns were also shared, most of them are valid for any children:

- supervision, presence, monitoring
- reputable and approved source
- not for excessive time
- filtered Internet

In addition to the ones above that were very commonly mentioned, some more specific warnings were also received. Someone pointed out that children with epilepsy should not use computer. Others mentioned the risk of becoming dependent or addicted to computers, especially games. It was also mentioned that using computer extensively for several different purposes may result in poor social interactions and an autistic child should get encouraged right to the opposite. What is important to point at is that no concern were raised that would claim that children

with autism are more vulnerable to dangers that are related to computers and the Internet than other children.

Further questions (and the responses given to them) unveiled that users do want

- much more help with social skills and communication skills
- more help with therapy, sensory and motor development, visualization and activity schedule
- some more tools for learning, language skills

Apart from that, they are indeed interested in virtual reality, co-operative games, robots and modelling, while it seems that it is never enough with entertainment.

A very important aspect from the prospective of this research was also questioned: this is the cooperation needs between different people that are involved in supporting a certain child on the autistic spectrum. The responses reflect a strong agreement in the following two statements:

**There are more than one person in charge to help a child, and  
It is a concerted effort to support children with autism**

The following statements were also more agreed than not:

**Communication is NOT easy among those people, and  
It is an issue sharing information among all those people.**

The last question asked was identical to one of the the final questions that were put forward in the above mentioned US survey, with the expressed permission of Ms Cynthia Putnam who kindly shared the full survey she performed with me.

**“In a perfect world, what type of software or technology (if any) do you think children on the autistic spectrum would really benefit from?”**

This last question was to provide a possibility for the respondents to tell about those wishes and requirements that could not be squeezed under any category in the previous questions. Answers given to this question are somewhat more open-ended and unrestricted so there are indeed a few interesting ones to highlight, apart from the categorized indications mentioned above. The ideas outstanding from the answers to earlier questions were:

- A personal diary, schedule, blog for cooperation between all the involved family members, assistants, therapists
- a development diary where the therapists log their notes and parents can response
- A robot they can make friend with

All the other answers were repetitions from the area of social skills, communication skills, language and learning aids. Looking at the evaluation of the same question from the Putnam-report: *“Answers to this question fell into three categories, goals (62%), access concerns (5%) and design suggestions (33%). Goals were further subdivided into three domains, (a) social/communication; (b) academic/school help; and (c) scheduling/organization.”*

## **3 Design Proposal**

### **3.1 Orientation**

The demand is confirmed to be present. The survey that was conducted also legitimised the topic of this research. There is still place for supporting applications, there is still need for better, easier, more versatile software that can help people to support and develop children with autism. Also it is important to highlight that teamwork capabilities, cooperation, information sharing are existing and valid requirement. This project especially focuses on delivering a solution that offers cooperative features. There are numerous special tools and methods that are widely used when dealing with autistic children. Most of them are based on visualization – trying to provide a visual aid for those everyday tasks that are easily adopted by an average child but sources of difficulties for a child on the autistic spectrum. Quite a few of these visual techniques can be migrated to computer applications. Using computer applications when creating and using these visual clues is a major help for those that are involved in creating and updating them. However, there is still an issue even with the computer-aided techniques. It is always a concerted effort that is needed to gain results.

Usually, more than one person is involved in the care of a child with autism:

- parents
- other members of family
- class teacher
- resource teacher
- special needs assistant
- speech therapist
- occupational therapist
- doctor
- etc. (e.g. nurses, foundation workers, social workers)

If any or some of these people use computer-aided visual aids when dealing with an autistic child, they will still need a way to share those tools. This sharing could be facilitated using some kind of networking. Unfortunately these people are usually not in the same building or organization, and do not have access to a common server. Setting up a proper client-server based solution with external access (like a VPN connection) is usually beyond the budgetary and technological limits of an average school or institute.

In line with the identified needs the goal is to find a way where those people who are involved in the education, development and care of children with autism can use software tools in a cooperative way, sharing their efforts and work more closely and efficiently.

### ***3.2 Why Cloud Computing?***

Cloud computing these days offers easy-to-use cooperative tool sets and methods for different purposes. These services are versatile, scalable and affordable. There is the possibility of using cloud computing to facilitate cooperation between the several parties that deal with children on the autistic spectrum by not only moving their visual techniques onto the computer but also making them widely distributed and shareable over the Internet.

- **Cooperation** – the aim is not only helping individuals in their support activity but to make cooperation easier among those that help an autistic child.

- **Networking** – to facilitate cooperation between individuals, we need to use networked computers and shared resources over the network.
- **Internet** – a local network is not enough. Not all the stakeholders are in the same organization. Parents, teachers, therapists – they are rarely on the same LAN.
- **Inexpensive Infrastructure** – while there are ways to connect to a local area network from outside, it is not likely that all schools and institutes can afford both the infrastructure and the support personnel to create and operate a VPN gateway on their LAN. A cheaper, more affordable infrastructure is needed that is available from any computer with an Internet connection, easy to operate and offers ready-made solutions for team work.
- **Customizable and versatile** – special functions and interfaces are required for special purposes. The solution must offer customization via API's.

Cloud computing is found to be a good solution that ticks all the boxes.

### ***3.3 Evaluating Google Cloud***

Currently, Google is one of the few major cloud service providers. There are several basic services that are available free of charge, others can be used for a reasonable cost. Google provides a well documented API for all services. Several cooperation and teamwork services that are provided by Google can be used without any modification to support developing a child. Other services can be used to facilitate developing graphical front-end applications. The following Google services are considered being useful:

- **Blogger**: A personal diary, development records, notes about the child that can be shared between several readers and editors.
- **Google Mail**: E-mail, communication tool for all that are involved.
- **Google Calendar**: Scheduling tool, provides easy ways to create and maintain the daily/weekly schedule of a child. Calendars can be shared, invitations can be send for appointments.



- **Google Docs:** Document and data storing solution
- **Picasa Web:** Picture and photo storage place where the graphical representations for visualization tasks can be stored – and recalled from.
- **Google App Engine:** Hosting for custom web applications that can make use of accessing Google services, manage authentication and store data.

### **3.4 Requirement Summary**

Children on the autistic spectrum need visual clues to perform certain organization or scheduling tasks that are fairly obvious to those who do not suffer from ASD. A certain percentage of children with ASD do not use spoken language at all, they communicate using picture cards. Parents, teachers and other professionals use pictograms, picture cards, drawings or photos for prompting, reminding or helping communication. Each individual built up his or her own set of pictures, later they were shared between each other, books and publications were printed with thousands of pictures. Now such picture databases are available on the Internet, with various availability and licensing.

The framework should provide tools to organize user data, groups of users and an overall application domain. It should manage a hierarchical structure of personal, group and generic picture set (galleries), retrieving pictures based on keywords and the user. If the user has a picture in his or her own gallery, it should be returned, otherwise the one from the group gallery (if there is none, then the generic picture) is to be returned. Using these features, special graphical front-end applications can be created that make use of these functions and translate text information into pictures. The application prototype is a visual agenda (called **VisAge**) that uses Google Calendar for scheduling and managing the daily routine of a child but provides a graphical interface to display the different tasks and events to the child.

The VisAge – Visual Agenda application is a constructive answer on the question “*how to make cloud computing usable for people on the autistic spectrum and those who are supporting them?*”

### **3.5 Licence information**

Because of its nature, in this project several photos, images, pictures and pictograms are used. These are from the following sources:

- **Sclera.be Pictogram Library**, by Sclera NPO. Licensed under a Creative Commons Licence
- **Mulberry Symbol Set** from Straight-street.com. Licensed under a Creative Commons Licence
- **ARASAAC Symbol Set** from Catedu.es. Licensed under a Creative Commons Licence
- Photos and images from **Wikimedia Commons**. Licensed under a Creative Commons Licence
- Other symbols from the **Picto Selector** application (freeware), created by its author, declared as freely distributable.
- Photos taken by myself (the author of this report), declared as freely distributable

Since the majority of the resources used are licensed under the Creative Commons Licence, I hereby declare that the application developed within this project is a non-commercial application. It is not and will never be sold or licensed for money.

## **4 Infrastructure Design**

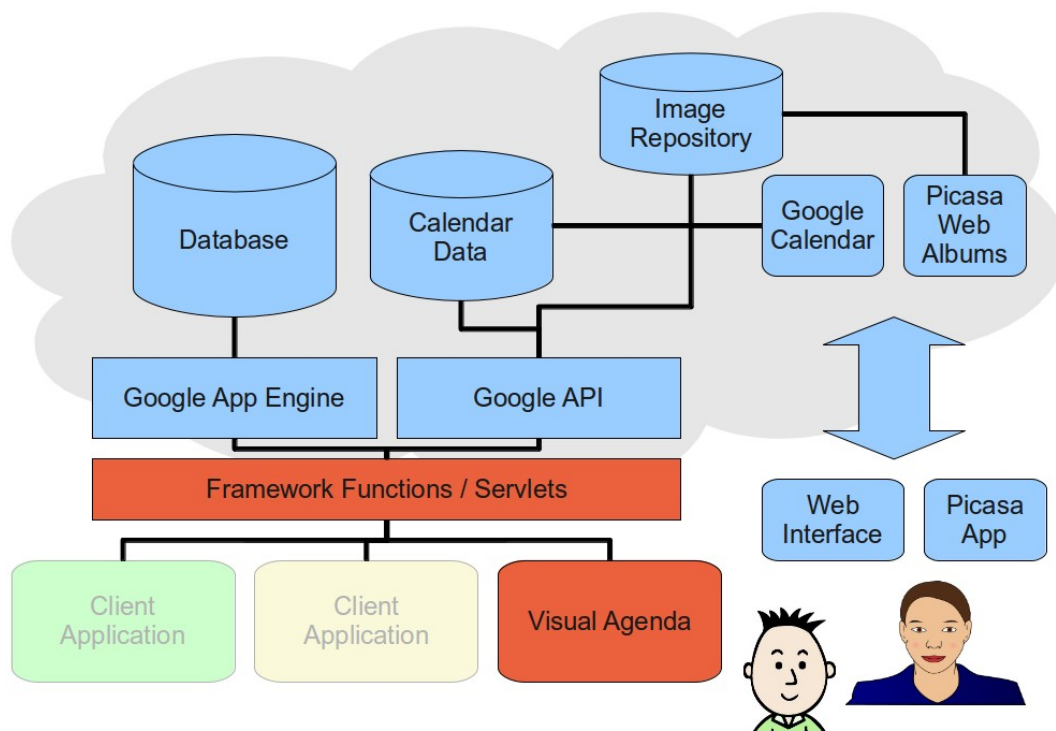
### **4.1 Environment**

The application is a single web project, hosted on Google App Engine. Google App Engine is a web application hosting service, provided by Google. It is a distributed system for running web applications which offers scalable web serving infrastructure. Up to 10 web applications can be hosted free of charge with a Google account. There are quotas set up for CPU time, data storage, bandwidth and some other measures. However, the rate of service that is achievable within the limits of free hosting is more than enough for demonstration, testing and also for use up to approximately 50 concurrent users.

Google App Engine supports Go, Python and Java as programming language, this application is written in Java.

The development environment was *Eclipse (Helios SR1)*, with *Java Enterprise Edition 6 Update 1* and *Google App Engine SDK 1.5.1*. Development was performed under *Ubuntu Linux 10.04*.

## 4.2 System Components



*Drawing 2: System components*

### Image Repository & Picasa Web Albums

Obviously, one of the most important part of the application is the image repository. This is the place where the image files are stored. Currently, Google provides 1 GB storage place for each registered user. Knowing that the size of one pictogram is approximately 10 kB in average and even a rich photograph with the recommended dimensions (maximum 300x300) does not exceed 150 kB in size, this capacity is sufficient to store literally thousands of pictures. In Picasa Web Albums images of a user are organized into albums. From the application's prospective, pictograms are

grouped in categories. The framework itself do not limit the number of categories. For the Visual Agenda application the pictograms are in 5 categories:

- Activity
- Objects
- Terms
- People
- Locations

One category in the application is matched to one album in Picasa Web Albums. Picasa images are stored on Google's servers and are available for custom applications via Google Picasa API.

### **Picasa Application**

The Picasa application is a photo manager software from Google. It is available for Windows, Linux and Mac. It is a great help in managing photos (or pictograms in our case), organizing them into albums and synchronize these albums with Picasa Web Albums. While there is a web interface also provided for Picasa Web Albums, it is much more convenient to use the client application.

### **Calendar Data & Google Calendar**

Google Calendar is an ordinary online calendar, which provides the usual functionalities for scheduling events and appointments. It features repeat patterns, inviting others, assign locations and adding text contents. Calendar data are stored on Google's servers and are available for custom applications via Google Calendar API.

### **Calendar Web Interface**

Google Calendar is available via web, using any modern browser on any most operating systems that have a graphical interface. All functions mentioned above are available via the web interface, it is the preferred method to access the calendar.

### **Google API**

The Application Programming Interface of Google services (Google API) provides access to data and functions of different services. Google uses Atom feed, which is

an XML-based format for web feeds. The application uses such data feeds of Picasa Web Albums service to retrieve pictograms from Image Repository and data feeds of Google Calendar service to retrieve the events from the user's calendar.

### **Google App Engine & Database**

The application is hosted by Google App Engine. It provides the running environment, the application administration, debugging and logging possibilities. It is also possible to schedule certain tasks and run background processes. The database that is provided by Google (App Engine Datastore) is a storage for objects. Unlike a traditional relational database, this one stores object entities. The SDK provided by Google contains implementations of Java Data Objects (JDO) and Java Persistence API. In the application these interfaces were used for data storage in the cloud.

### **Framework Functions & Servlets**

These functions are part of the application, however, they can serve as a basis for other similar applications. The framework servlet objects are placed in an own package so they can be re-used with another application. Functions of the framework are:

- User management
- User group management
- Pictogram retrieval

### **Visual Agenda**

This application is a proof-of-concept demonstration. It reads the data feed from the user's calendar and using the pictogram retrieval function of the framework it displays a visual agenda that can be used when explaining or teaching an autistic child.

## **5 Application**

The Visual Agenda web application is developed using the Java programming language. The server-side functions are performed by Java servlets, the user interface functions are Java Servlet Pages (JSP). Client-side functions are performed using

JavaScript. In some cases, AJAX (Asynchronous JavaScript And XML) methods were used for the smooth user experience. Certain static pages are pure HTML files. For the styling, a Cascaded Style Sheet (CSS) was used.

## **5.1 Design**

### **5.1.1 User Stories**

Design of the application began by collecting the user stories. Based on our own experiences and by discussing with different professionals (resource teacher and therapist) the following stories were created as guidance for the design:

#### **(A) Managing the application**

*User stories of the application's main administrator(s)*

- The administrator create groups, assign group administrator to them and also delete groups.
- The administrator can view the complete user list of the application.
- The administrator can manage all users of the application, in terms of refusing application of new users, redirecting users to a different group, deleting existing users. The application administrator cannot add a user to any group, it is the group administrator who can accommodate a user into his or her group.
- The administrator manages the generic, application-level picture albums. (This function is performed using Picasa, outside of the application)
- The administrator has the possibility to force updating the picture database from Picasa Web Albums.

#### **(B) Managing a group**

*User stories of a group administrator*

- The group administrator can set and change the description of the group
- The group administrator manages the common, group-level picture albums. (This function is performed using Picasa, outside of the application)
- The group administrator assigns the picture albums to categories.

- The group administrator has the possibility to force updating the group's picture database from his/her own public Picasa albums.
- The group administrator can view the list of users in his or her group.
- The group administrator can accommodate a user who applied to join the group.
- The group administrator can refuse a new user's application.
- The group administrator can remove an existing user from the group.
- The group administrator cannot delete users, removed users become orphaned and can be redirected to other group by the application administrator.

### **(C) New user application**

*User stories of a new user who wants to use the application*

- New users land on a welcome page. New users are not allowed to use the application without being registered.
- New users can apply for joining an existing group. The group administrator gets notified by e-mail about the new applicant.
- New users can apply for creating a new group and administering it. The application's administrator gets notified by e-mail about the new applicant.
- New users can only apply once. When they return to the application while their application is still pending, this information is displayed and no further application is possible
- Refused users cannot re-apply until they are explicitly deleted by the application's administrator
- Orphaned users cannot use the application until they are redirected to another group and accommodated by this group's administrator.

### **(D) Editing user's profile**

*User stories about editing user's own details*

- User can review his/her own details retrieved from Google, like User ID, e-mail address, nickname and the authentication domain. No changes are possible to these data.

- User can view and change his/her full name as stored in the application's database
- User can view and change his/her nickname as stored in the application's database
- The user manages his/her private, user-level picture albums. (This function is performed using Picasa, outside of the application)
- The user assigns the picture albums to categories.
- The user has the possibility to force updating the picture database from his/her own Picasa albums
- The user needs to grant access to Picasa Web Albums so his/her own pictures can be retrieved.
- The user needs to grant access to Google Calendar so his/her own agenda can be retrieved

## **(E) Displaying visual agenda**

### *User stories on viewing a daily schedule in pictograms*

- The user wants to see the overview of the daily agenda. Each event is represented by one single pictogram on a linear time-scale.
- The user can step a day forward or backwards to see yesterday's or tomorrow's agenda.
- By default, the user wants to land on the current day. The time-scale shall scroll to the first event in the day.
- The user clicks on an event's pictogram to see its details displayed on the main part of the screen.
- When clicked on an event's pictogram, the details in plain text are displayed on the top.
- When clicked on an event's pictogram, the details translated into pictograms are displayed in the centre.

### **5.1.2 General Considerations**

In order to use the application, the user must be signed in with his Google account. The user must grant access to his/her Google Calendar, otherwise the application



cannot get the events feed from Google API so no events will be displayed in the agenda. If the user wants to benefit from having his/her own pictures displayed when available, access to Picasa Web Albums must be granted.

User shall be provided with a Sign out function so (s)he can safely log out from the application and prevent others accessing his/her private data.

The name of the current user is displayed in the header so it is always visible.

### 5.1.3 Pictogram translation

The guidelines on how to translate plain text into pictograms is agreed as follows:

The program shall first look for a match in the user's own albums. If there is no matching picture, then look for one in the group's common album which the user belongs to. If there is no matching picture either, then look for one in the application's albums.

When searching for match, the primary hint shall be the description of the picture. If no match found with the description then look into the tags. If no match found with the description then look into the file name.

When interpreting the **Subject** field of an event, perform the search in the **Activities** category. This will be the single-pictogram representation of the event in the time scale.

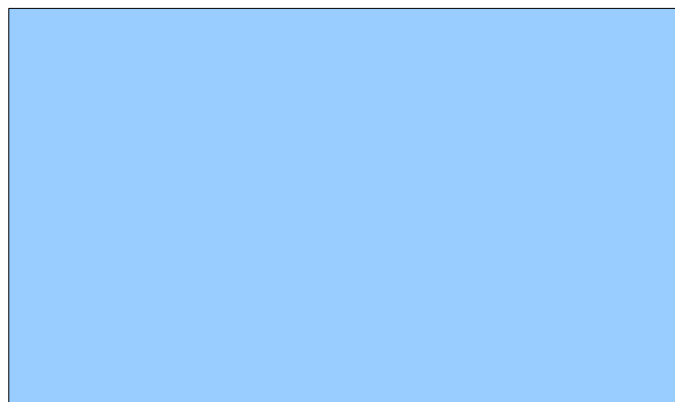
When interpreting the **Where** field of an event, perform the search in the **Locations** category.

When interpreting the **Guests** field of an event, perform the search in the **People** category.

When interpreting the lines of text in the **Description** field of an event, each line shall be interpreted as an imperative **sentence**. The algorithm to use is the following:

- Remove any punctuation.
- Remove words like a, an, the, this, with, in, on, to, etc.
- If the sentence begins with “Do”, remove it from the sentence and mark it as “Stressed”.

- If the sentence begins with “Don't” or “Do not”, remove it from the sentence and mark it as “Unpermitted”.
- Take the first word in the sentence and look for a matching pictogram in the Activities category. If the sentence is marked as “Stressed”, place a green tick mark over the picture to highlight as good practice. If the sentence is marked as “Unpermitted”, place a red cross over the picture to highlight it as being forbidden.
- Take the second word in the sentence and look for a matching pictogram in the *Objects* category, then in *People* category, then in *Terms* category
- Take the third word in the sentence and look for a matching pictogram in the *People* category, then in *Location* category, then in *Terms* category
- Take the fourth word in the sentence and look for a matching pictogram in the *People* category, then in *Location* category, then in *Terms* category, then in the *Objects* category.



*Drawing 3: Translating Calendar event into pictograms*

## 5.2 Implementation

### 5.2.1 Data Objects

As mentioned, the data storage of Google App Engine is storing object, not records.

Objects of the following classes are stored:

#### (A) User Data (UserData.java)

*Objects of this class store all user-related data that are necessary for the application.*

#### **Fields:**

String <b>userId</b>	This is the unique Google user ID of the user. Used as a key in the database
com.google.appengine.api.users.User <b>gUser</b>	The user object, that is returned by the App Engine API after login. Contains only a few basic data of the user, from the Google Profile
String <b>fullName</b>	The full name of the user
String <b>nickName</b>	The nickname of the user, a short name to display
String <b>emailAddress</b>	The e-mail address of the user
String <b>userGroup</b>	The name of the group which the user belongs to
String <b>status</b>	The status of the user, either applied, refused, active or orphaned
List<String> <b>privAlbumCats</b>	This data structure stores the identification of the user's own Picasa Albums for different categories
List<String> <b>privAlbumIds</b>	
List<String> <b>accessServices</b>	This data structure stores the access tokens for different Google services
List<String> <b>accessTokens</b>	

**Methods:** (excluding the basic getter/setter methods)

HashMap<String,String> <b>getPrivatePictureAlbums()</b>	Returns a HashMap object with all the user's private albums by category
String <b>getPrivatePictureAlbum(String category)</b>	Returns the private album for a given category
void <b>setPrivatePictureAlbum(String category, String albumid)</b>	Sets the album id for a given category in user's private albums
String <b>getAccessToken(String service)</b>	Returns the access token for a given service
void <b>setAccessToken(String service, String accessToken)</b>	Sets the access token for a given service
boolean <b>isUserGroupAdmin()</b>	Returns true if the user is the administrator of the group (s)he belongs to

## **(B) User Group Data (UserGroupData.java)**

*Objects of this class represent groups of users*

**Fields:**

String <b>groupName</b>	Name of the group. Used as a key in the database
String <b>adminEmail</b>	E-mail address of the group owner / administrator
String <b>ownerUserId</b>	The user ID of the owner / administrator
List<String> <b>commonAlbumCats</b>	This data structure stores the identification of the group's own Picasa Albums for different categories
List<String> <b>commonAlbumIds</b>	

**Methods:** (excluding the basic getter/setter methods)

HashMap<String,String> <b>getGroupPictureAlbums()</b>	Returns a HashMap object with all the group's albums by category
String <b>getGroupPictureAlbum(String category)</b>	Returns the group's common album for a given category
void <b>setPrivatePictureAlbum(String category, String albumid)</b>	Sets the album id for a given category in user's private albums

Unfortunately, the data store has limitations on the field types that can be used in objects that are stored in it. One of these limitations is that it can only be either a primitive data type, or a single object, or one of the few supported *Collection* object, like List. It does not support classes like *HashMap* or similar to store key-data pairs. Both the private album list of the user and the access token store would ideally be a *HashMap*. Because of the limitations, they are realized as two separate lists and the getter/setter methods translate them into *HashMap* or mimics the map-like behaviour.

### (C) Photo Data (PhotoData.java)

*Objects of this class store the required data of pictogram images*

#### **Fields:**

String <b>photoId</b>	Unique identifier of the photo object. Used as a key in the database
int <b>type</b>	The source type of the photo: (0 = user; 1 = group; 2=application)
String <b>category</b>	Pictogram category
String <b>keyword</b>	The description of the picture, the main keyword field when searching
String <b>filename</b>	The name of the original picture file, without the file extension
String <b>taglist</b>	Contains the tags associated with the picture
String <b>owner</b>	Describes the owner of this picture, for type 0 it is the user ID, for type 1 it is the group name, for type 2 it is the string constant "Application"
String <b>photoUrl</b>	This is the URL of the picture that can be put in the src attribute of an IMG element

#### **Methods:**

There are only the basic getter/setter methods for this object.

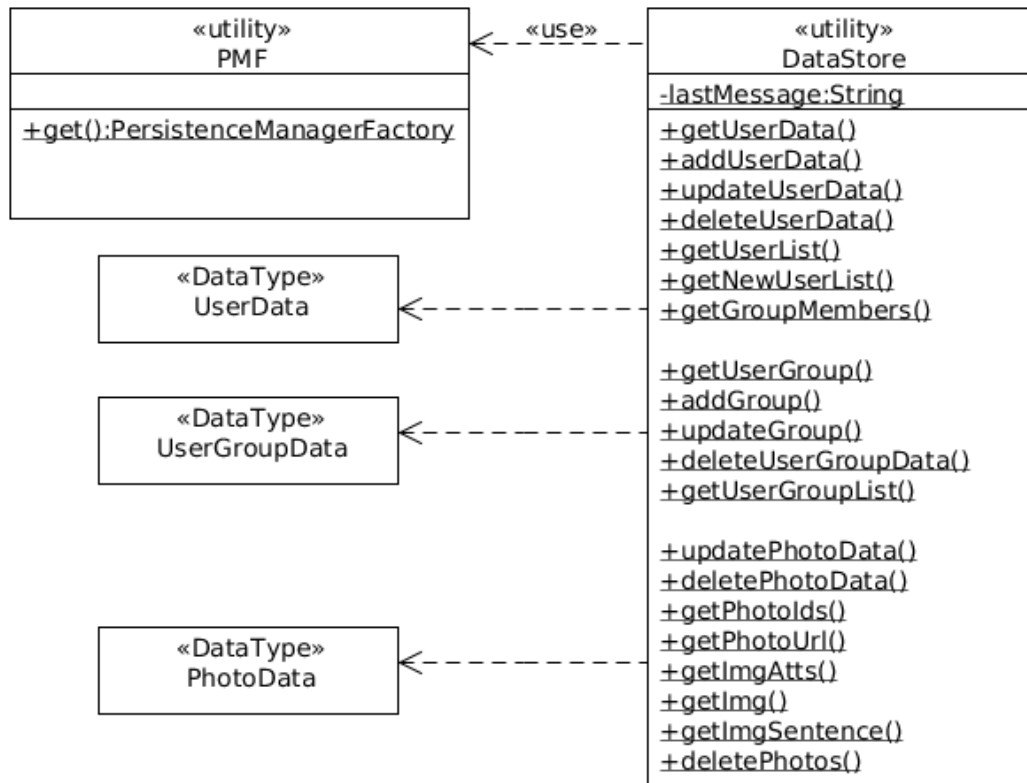
### **5.2.2 Data Access Objects**

Accessing data objects that are stored in Google's data store is a relatively straightforward procedure. It implements the Java Data Object and the Java Persistence API interfaces so there is a way to access data with high level methods. There is a low-level access mechanism as well but it is more convenient to use JDO

and JPA interfaces. The data structure used in this application is not very complicated so there is really no need to bother with low-level access.

The choice was the Java Data Objects, simply because it is better documented on Google Code (the Google App Engine and Google API documentation portal). The application uses the objects and methods declared in the JDO interface to access data. The interaction between the application and JDO is managed by an instance of *PersistenceManager* class. According to the recommendation in Google's documents, a singleton wrapper class is used to return a *PersistenceManager* instance – it is a timely procedure to initialize the *PersistenceManagerFactory* instance which creates the manager object according to the configuration settings. The *PMF* class is this wrapper, with only one static method, *get()*.

Another utility class was created to perform all the database-related functions. It is called *DataStore*, it only has static methods to serve for data manipulation and data query purposes.



Drawing 4: Data Access Classes

The static methods in *DataStore* are called every time when data access is needed. There are a group of methods that deal with *UserData*, they make it possible to retrieve user information by user ID (*getUserData*), to add a new user to the database (*addUserData*), to update user information (*updateUserData*) and to delete a user (*deleteUserData*). The remaining 3 methods return *List<UserData>* collections, either the complete user list (*getUserList*), the list of newly applied users (*getNewUserList*) or the members of one group (*getGroupMembers*).

The second group of the methods belong to the *UserGroupData* objects in the database, they serve to retrieve, add, update and delete records of this kind. There is also a way to get a list of all groups.

The third group of methods are the ones that manage *PhotoData* objects in the database. The *updatePhotoData* either adds or updates a picture object. While it was necessary to distinguish between adding and updating users and user groups, it does

not really matter if the picture data was added or only updated. There is no editing possibility of picture data in the application itself, they are only updated from the Picasa Web Albums. The *deletePhotoData* method deletes one record while the *deletePhotos* deletes all photos from one owner. The method called *getPhotoIds* returns only the key fields of the photos that match the given criteria. Returning only the key instead of the whole object is a faster procedure and costs less CPU time. The remaining methods are the ones that are called for the picture translation functions. The core method is the `String getPhotoUrl(String keyword, String category, String userId)`. The purpose of this method is to return with the best match found for the given keyword within the specified category for the actual user. The method performs the search according to the specification detailed earlier: first in the user's album, then in his/her group's album finally in the application's album. Within the albums, first by the keyword, then by the tag list, finally by file name. This method returns null, if nothing could be found, or returns the URL of the pictogram which best matched. The next method, *getImgAtts* is built on top of the previous one. There are two differences: (1) it not only returns the URL but the complete attribute set that can be pasted into an IMG tag: the *src*, the *height* and the *title* attributes are populated. (2) it always returns a valid result, never returns with null. If nothing was found, it retrieves the pictogram called “unknown” from the Terms category – if nothing else, the application's album does contain a pictogram with such keyword. The third in the row is *getImg*, which calls the *getImgAtts* method but returns the whole IMG tag. Finally, the *getImgSentence* method returns a DIV tag that contains the images (IMG tags) representing the sentence with pictograms.

### 5.2.3 PictoManager

The class called *PictoManager* is the interface between the application and the Picasa Web Albums. Instance of this class is used when user authentication is required to connect to the web albums. This class also contains static methods to perform functions where user authentication is not required.



PictoManager
- <u>PHOTO_FEED_URL</u> : String - <u>ALBUM_FEED_URL</u> : String -userData: UserData -groupUserId: String -appUserId: String -userAlbums: HashMap<String, String> -groupAlbums: HashMap<String, String> -appAlbums: HashMap<String, String> -myService: PicasawebService
+updateAppPictoDatabase() +updateGroupPictoDatabase(String groupName) +getPublicPictoListFromUrl(String sourceUrl)  +PictoManager(UserData userData) +refreshData() +updateMyPictoDatabase() +getPictoListFromUrl(String sourceUrl) +getPictoList(String category, String source)  +getAlbumListForUser(String userId) +getAlbumListOf(String source) +getAlbumListForUser()  +searchForPictos(String sourceUrl, String keyword) +getMatchingPictos(String keyword, String category, String source) +getMatchingPictos(String keyword, String category) +getMatchingPicto(String keyword, String category)

*Drawing 5: The PictoManager class*

The constants *PHOTO\_FEED\_URL* and *ALBUM\_FEED\_URL* hold the skeleton for the corresponding feeds so when using them, only the user ID and/or the Album ID need to be replaced. In the case the URL changes in the future, only these constants are to be updated.

When using an instance of this class, then the fields hold the details related to the current user: The user identification (*userData*), the group identification (*groupUserId*), the album lists (both user-level and group-level). For easy reference, the application-level details are also stored with the object.

The *getAlbumList()* method and its siblings return the list of available albums for a given user, for the current user or from a certain source (like user, group or

application). These functions are used at group and user profile set-up, where the user is asked to choose from his/her own albums for different categories.

The *getPictoList()* methods – also the *getPublicPictoListFromUrl()* static method – return all the photo entries from one Picasa album.

There are two different approaches that can be used by the framework to retrieve pictures from Picasa Web Albums for a given keyword. The first approach is performing a search using the Picasa API and retrieve the matching pictogram based on the search results. This method is a pure “mashup” approach, with no local data storage. Once the need arises to translate a keyword into pictogram the application can turn to Picasa API, run a search and return the matching image. The advantages of this approach are: result is always up-to-date, no database background is necessary, very little overhead, lean application. Unfortunately, there are also disadvantages: this method is rather time-consuming, especially considering the algorithm that was described in the requirements section: There are 3 levels of search (description, tags, file name), 3 sources of search (user, group, application) and if working with sentences, up to 4 consecutive keywords are sought for. So in worst case,  $3 \times 3 \times 4 = 36$  search requests are processed for one image sentence. The time needed for this amount of API transactions when tested varied between 30 and 80 seconds. It is clearly not an option to use this approach for the VisAge application but it is well usable when only a few searches need to be performed.

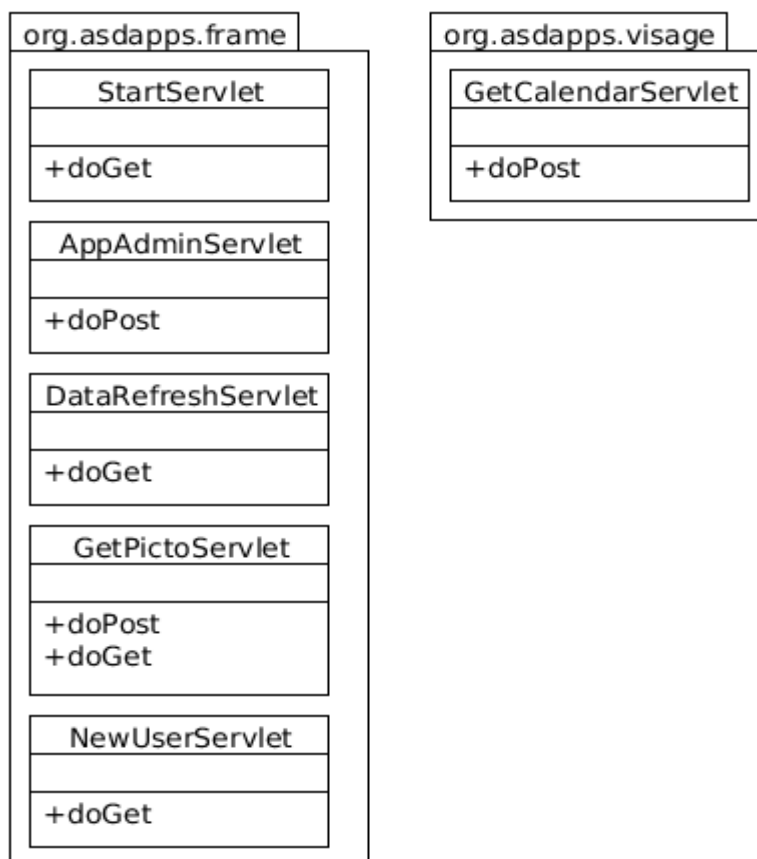
The second approach is to maintain a local database of *PhotoData* objects. All searches are performed using the database functions and when the proper object is found, take the URL stored with the picture. Calls to Picasa API are only used when updating the database. The advantage of this method is the fast speed that can be achieved, the drawback is the overhead of database management and the regular updating. Because VisAge displays several pictograms for one event and displaying event shall be relatively fast, the second method is used.

*PictoManager* class provides functions for both approaches. The three different *updateAlbum()* methods retrieve all the photo entries from all the albums of a given owner (user, group or application) and store/update them in the database. The last group of methods serve the “direct search” approach. The method called

*searchForPictos()* calls Picasa API and searches for photo entries by a keyword in one album. *getMatchingPicto()* and its siblings call this method one or several times to get the picture translation they need.

## 5.2.4 Servlet Classes

In the background, a few Java servlets provides administrative and other functions for the application. They do not produce any output that the user would see directly (except the *GetPictoServlet* which can be used by the administrator for certain service functions that return HTML content but it is not intended for the ordinary user). All servlet objects have their own well-defined functions, 5 of them belong to the frame package, they can be re-used for other applications, while the *GetCalendarServlet* class belongs to the visage package.



*Drawing 6: Servlet Classes*

## **(A) StartServlet**

*Purpose: administrative and housekeeping functions when an application session starts.*

When the application is started, the welcome page (*index.html*) is displayed. By clicking on the “Start” link, this servlet is called (GET). As a first step, it checks if the user is logged in with his/her Google account. If not, then redirects to a login page. If there is a logged-in user, the program tries to retrieve the user's details from the database. Known active users get redirected to the landing page of the application, while unknown or not active users land on the guest page. The *UserData* object gets stored in the session.

## **(B) NewUserServlet**

*Purpose: sending message about a new user application*

When a new user applies for registration on the guest page, this servlet is called (GET). All it does is composing and sending an e-mail message on behalf the new candidate to the corresponding group administrator or the application administrator, depending on whether the new user wants to join to an existing group or to start a new group.

## **(C) AppAdminServlet**

*Purpose: background functions for application and group management*

This servlet is the server-side processing object for the AJAX calls (POST) from the client-side JavaScript. Depending on the *action* parameter passed on to it, it performs different actions and returns plain text or HTML content which is then inserted into the page on the client side.

When called from the application's administration page, it can:

- return list of candidates to become group administrators (*getcandidates*)
- return the name a new user wanted to give to a new group (*getgroupsuggestion*)
- return the member list of one group or all groups (*getuserlisting*)
- delete a user (*deleteuser*)

- refuse a user (*refuseuser*)
- redirect a user to another group (*redirectuser*)
- queue a request to update application's photo database (*updateappalbums*)

When called from a group's administration page, it can:

- return the member list of this group(*getuserlisting*)
- refuse a user (*refuseuser*)
- accept a user's application to the group (*acceptuser*)
- remove a user from the group (*removeuser*)
- queue a request to update the group's photo database (*updategroupalbums*)

When called from a user's profile page, it can:

- queue a request to update the user's photo database (*updateuseralbums*)

The value of the *action* parameter is shown in parenthesis.

## **(D) GetPictoServlet**

*Purpose: various pictogram operations*

This servlet can serve as server-side object to serve AJAX calls (POST) from JavaScript. Depending on the *action* parameter it returns either an IMG tag with the pictogram translation of the specified keyword within given category; or returns a DIV tag with an image sentence. The same servlet is called (GET) from the application administration page, to provide service functions that helped during test and development, but can be useful later for hand-made queries against Picasa Web Albums.

## **(E) DataRefreshServlet**

*Purpose: updating PhotoData objects in database from Picasa Web Albums*

This servlet is never called on-line, only queued in the background task queue or invoked as a scheduled job. It is performing the update of picture database from Picasa Web Albums.

## **(F) GetCalendarServlet**

*Purpose: background functions to support calendar display*

This servlet is the server-side processing object for the AJAX calls (POST) from the client-side JavaScript in the application's main page (Calendar). When called with *action* parameter *geteventsforday*; it returns the list of events for a given day, so they can be displayed in the day schedule. When the *action* parameter is set to *geteventdetails*; it returns the translated version of the event with all the details.

### **5.2.5 Front-End Pages**

Being a web application, the user interface of the application is constructed from web pages. Most of them are Java Servlet Pages (.jsp), with only two exceptions: the welcome page and the about page are static HTML pages. Below is a short summary about the different front-end pages and their functions:

#### **Welcome page (index.html)**

This is a static HTML page with the sole function of telling the user what is the application (s)he arrived to and providing the link to start (which invokes the *StartServlet*).

#### **About page (about.html)**

Another static page with just a few basic information about the application itself and the references for the sources I used.

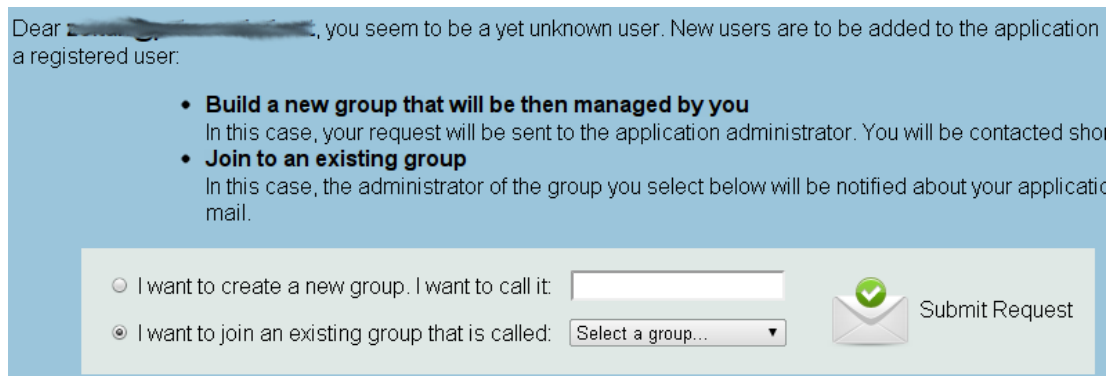
#### **Contact page (contact.jsp)**

There is a possibility to send an e-mail to the application's administrator using this page. This page is available for all users regardless of their status and also for not registered users so they can enquiry about the application.

#### **New User Application page (newuser.jsp)**

This is the page that welcomes new users who are not registered in the application yet. At the first visit, the user can decide whether (s)he wants to join an existing group or to start a new group. If (s)he aims to join a group, then (s)he needs to select

the desired group and then submit the request. If his/her intention is to start a new group then the selected name is to be filled in and then the request can be submitted.




Dear [redacted], you seem to be a yet unknown user. New users are to be added to the application as a registered user.

- **Build a new group that will be then managed by you**  
In this case, your request will be sent to the application administrator. You will be contacted shortly.
- **Join to an existing group**  
In this case, the administrator of the group you select below will be notified about your application via email.

☐ I want to create a new group. I want to call it:

☒ I want to join an existing group that is called:

 Submit Request

*Drawing 7: New User Application page*

Once a request is submitted, the user gets stored in the database with a status “applied”.

Not only new users get redirected to this page, all users with a status other than “active” (applied, refused, orphaned) land on this page where a short notice warns them about their current status.

### **Application Administration page (appadmin.jsp)**

This page provides access to those functions that are to be performed by the application administrator(s). An administrator can – using this page – manage groups: create a new group, delete an existing group, assign administrator to a group. It is also possible to manage users: (s)he can list the users of any group, can refuse applied users, redirect users to another group, delete users. The application administrator cannot accommodate new applicants into an existing group – this is the right of the group administrator.

On this page the administrator can force an update on the application's picture database. Finally, certain service functions that make possible interacting Picasa API are also available from this page – primarily for debug and development purposes.

### **Group Administration page (groupadmin.jsp)**

This web page facilitates group management within the application. The group's administrator can assign the Picasa albums of the group to the different categories. It




is possible to force an update of the group's picture database from Picasa Web Albums. A group description can also be specified on this page.

The other main function is the group's user management. The group administrator can list the active, the applied and the refused users. (S)he can accommodate new applicants in the group or refuse their application. It is also possible to remove a member of the group. Removed users become orphaned – group administrators cannot delete users.

### **User Profile page (getinfo.jsp)**

On this page the user can edit his/her own preferences. There are 3 main purposes of this page: (1) The user can give some details about him/her, like full name or nickname. (S)he can also check some details from the Google account: the user ID, the e-mail address, the nickname as stored in the Google profile and the authentication domain. (2) A very important function on this page is granting access to the Google Calendar and the Picasa Web Album services. The application can only access the data in our private calendar if we explicitly grant access to it. Similarly, the only way to retrieve private pictures or even the private album list from Picasa is via an access token that the user must provide by giving access for the application to query his/her private data. (3) If the user wants to use his/her own picture set in the application, (s)he need to assign private albums to the different categories. Album names can be selected from the drop-down list – provided that the access has been already granted. It is also possible to force an update of the user's picture database from Picasa Web Albums.

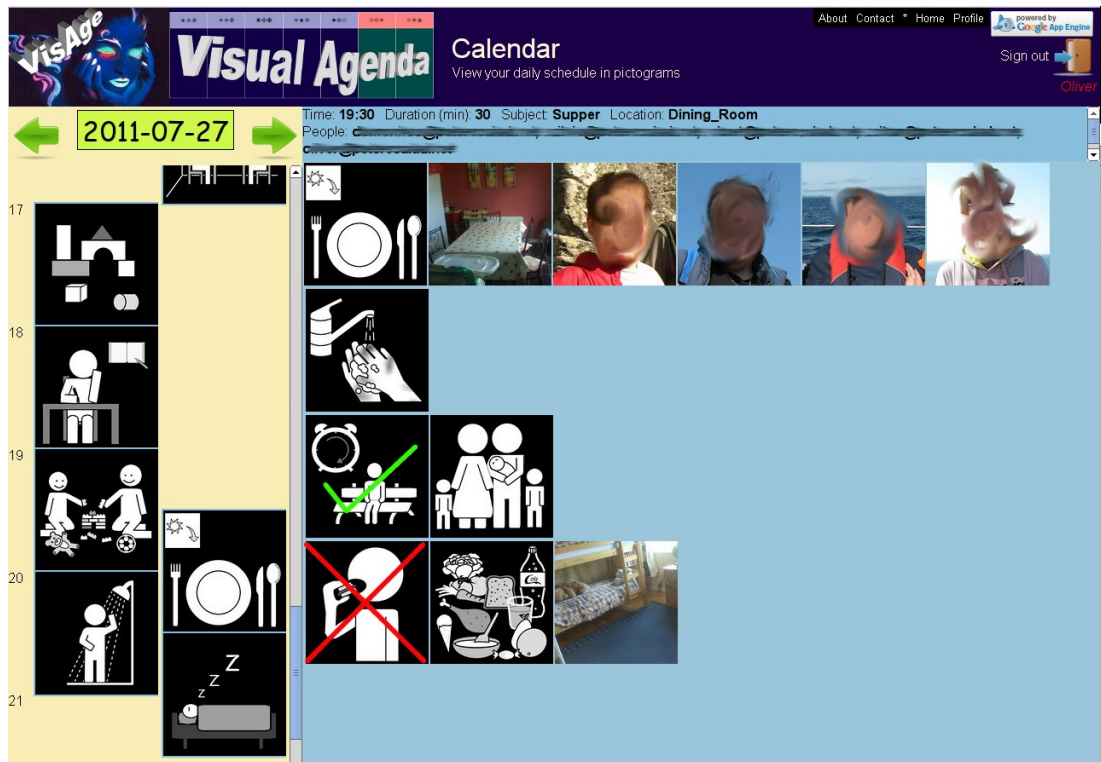


Google User ID:	114760595618461368732	Full Name:	Robert Peter
Google User Nickname:	robert@petercsalad.net	Nickname:	Robi
Google Authentication Domain:	gmail.com	E-mail address:	robert@petercsalad.net
Google Calendar authentication:	Granted	grant access to	
Google Picasa authentication:	Granted	grant access to	
Please select your private albums:		***	
My <b>Activities</b> album:	MyActivities ▼	Submit Changes 	
My <b>Locations</b> album:	MyLocations ▼		
My <b>People</b> album:	MyPeople ▼		
My <b>Terms</b> album:	MyTerms ▼		
My <b>Objects</b> album:	MyObjects ▼		
<input type="button" value="Update Albums in Database"/>			

*Drawing 8: User Profile page*

### **Calendar page (visage.jsp)**

This is the most important web page in the application – in fact everything else only exists to make it possible to display this page. On this page the details about the events that were retrieved from the Google Calendar data feed are displayed as pictograms. There are three sections on this page. On the left, there is a horizontal time-scale, where each event of the day is represented by one pictogram. This is the daily schedule, an overview of the day. When someone clicks on one of the pictograms on the left, the details of this particular event gets displayed on the right. In a narrow band on the top there is a text summary about the event, while below that there is the pictorial representation of the event details.



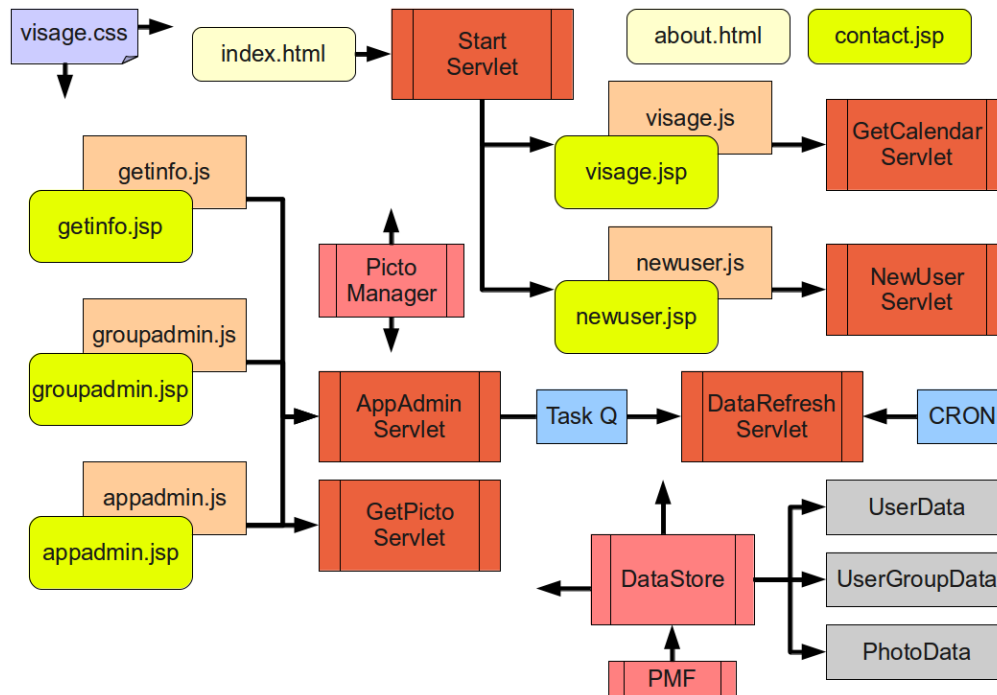
*Drawing 9: Visual Agenda on the Calendar page*

This is the interface which is intended for use with the autistic child. The adult can look at the text summary above or the tool-tip text of each pictogram for verification. The vertical time-scale on the left provides the daily overview and the main section is the detailed explanation of the selected event. In the picture above the event selected is the evening meal – supper. On the right the details are

- the activity itself (supper), from the application's Activities pictogram set
- the location (dining room), from the user's own Locations album
- the others who are involved in this event (the user's family members), from the group's People album
- The lines, interpreted as sentences, from the description field: Wash hands / Do wait for the family / Do not eat food in the bedroom.

### 5.2.6 System Overview

After introducing all the components of the Visual Agenda application, below there is an overview of the system as a whole.



### Drawing 10: System Overview

### 5.3 Testing

Due to the nature of such projects, the development was performed only by myself. While extensive component testing was carried out during the development work, there were no code reviews or other formal testing involved.

Once the development was finished, a test session was organized, involving a boy (Oliver) with Asperger's syndrome, his parents and the therapist who is working with him on a regular basis. They all formed a group and started to use the application. The test was a functional test using black-box approach. Participants got a brief summary on how the different Google services work and on how to use the VisAge application. I was also involved in the testing – as I am the father.

The test contained the following steps:

### (A) Preparation

Several pictograms were uploaded to the application-level albums to have enough expression to choose from.

The group-level albums and one private album of Oliver also got at least one picture in them

While organizing the pictures in Picasa, their description, tags and file name were edited and notes were taken about the expressions used.

In the application itself, user registration were completed, also the user profiles were edited and access was granted to both Picasa and Google Calendar.

### **(B) Schedule**

Calendar entries were created in Google Calendar for the test. Several different combinations were tried, like single appointments, repeating appointments (daily and weekly), standalone events (one person only), group events (where Oliver was invited to), group events owned by him (where others were invited to). The calendar entries varied in content as well: from the one that only had a subject line to others that had 5 lines of text in the description field.

When filling in the fields in the Calendar, expressions were selected to sample all combinations below:

- only available on application level
- only available on group level
- only available on user level
- available on both group level and application level
- available on both user level and group level
- available on all levels

The aim was to confirm that the application always selects the correct picture as described in the requirements.

### **(C) Verification**

Once the preparation work was done and the schedule was created in Google Calendar, the application was started and using the Calendar page we checked the displayed visual agenda against the original Google Calendar entries in a web browser. The verification covered the following angles:

- all events show up in the agenda, being single or repeated, standalone, group event owned by the user or group event where the user is invited to.
- Time and duration is indicated properly
- Pictogram translation of different fields and the sentences are using the proper categories
- The pictograms were chosen from the correct level of the customization hierarchy (user over group, group over application)
- Keywords were found equally in description field, in tag list and in file name.

## **5.4 Release**

The first release of the application is available now for use. Uploading pictograms with descriptive names and tagging is still an ongoing background activity. It is not used actively yet, mostly because of the summer vacation. A pilot group (led by the therapist who was the main consultant) will start using it from September onwards and provide feedback on the application. The application is available on the following URL: <http://cloudvisage.appspot.com>

## **5.5 Future enhancements**

The application as released in its current version is capable to fulfil all the requirements that were collected during the design phase. However, during the development and especially on the test there were a few points identified where there is room for further development. In the future, the following possible enhancements will be checked and probably implemented:

Help the user by providing the list of expressions that the application currently understands. As the application is not supposed to act as an entity with artificial intelligence which understands a human language like English, the user is limited to use a certain vocabulary when creating events. It would be very helpful to provide some kind of index function, or a lookup interface where the user can quickly get information about the expressions that (s)he can choose from.

A further enhancement for the feature described above would be a visual composer, where the event could be composed directly by using pictograms. This event would

be saved then to Google Calendar where further administration could be performed. (repetition settings, inviting guests, etc.)

The current interpretation method of sentences does not cover all situations. It is primarily aimed at warnings and imperative sentences. There are other kind of sentences and word lists that should be treated differently. A more sophisticated sentence interpretation that can adapt to more than one structure is necessary for more versatile use.

In the current version, a single administrator manages all application-level pictograms, one person from a group manages group pictograms and the average user only has access to his/her own images. As it usually happens in such communities, there is always someone who finds or creates new pictograms. As a first step, it gets added to the user's own album. However, if (s)he wants to offer it to the group or to the application, there is no easy way to do so. There should be a way to donate one or more pictures from a user's set to a group or to the application, where the administrator could accept or reject on the push of a button. Accepted pictograms would become part of the community set.

Finally, the printing of the schedule arose as a future enhancement request. Even though one of the major aims were to make paper-based schedules obsolete and use the on-line visual agenda as a paperless solution, certain habits do not change. Time to time user wants to print it out. In its current form, the Calendar page is not very printer friendly, so if printing is really required, a printer-friendly version is to be constructed.

## **6 Conclusion**

This research project was a great experience which resulted in an interesting learning curve about assistive techniques for myself and an exciting new tool for the involved professionals. During the background research I got in contact with several other developers who share all their work with others to help supporting children who suffer from different mental conditions. Most of them – like myself – started their work because of being personally involved. The local teachers and therapists I talked to during the design phase got also very excited about the new possibilities and gave

a lot of support. I am sure that this project will stay alive and the application will evolve with time.

Cloud computing is all around us and every day more and more people make good use of Internet-based applications. We can access our e-mails, calendar, documents, social network anywhere, at home, on the go, in a hotel, at the workplace. All we need is a browser and the possibilities are endless. Also the increasing popularity of hand-held devices with mobile Internet access points to the direction that all our personal data will be migrated to the clouds. On the other hand, applications in the cloud are designed with ordinary people in mind. With this project it was proven, that by adding just a little to extend these applications they can become great tools in supporting children on the autistic spectrum.

# References

- [1] Apeldoorn, G., AutiPlan - picto-planning made easy. Available at: <http://www.autiplan.com/> [Accessed April 27, 2011].
- [2] APPI, 2000. *Diagnostic and Statistical Manual of Mental Disorders, 4th Edition, Text Revision (DSM-IV-TR)* 4th ed., United States: AMERICAN PSYCHIATRIC PRESS INC (DC).
- [3] Attwood, T., 2006. *Asperger's syndrome□: a guide for parents and professionals*, Philadelphia: Jessica Kingsley Publishers.
- [4] Burke, M., Kraut, R. & Williams, D., 2010. Social use of computer-mediated communication by adults on the autism spectrum. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*. Savannah, Georgia, USA: ACM, pp. 425-434.
- [5] Domican, L., Grace App for Autism on iPhone, iPad and iTouch. Available at: <http://graceappforautismoniphone.blogspot.com/> [Accessed June 8, 2011].
- [6] Frost, L. & Bondy, A.S., 1999. *PECS training manual□: the Picture Exchange Communication System* [1999 ed.], Cherry Hill NJ: Pyramid Educational Consultants.
- [7] Grynspan, O., Martin, J.-C. & Nadel, J., 2005. Human computer interfaces for autism: assessing the influence of task assignment and output modalities. In *CHI '05 extended abstracts on Human factors in computing systems*. Portland, OR, USA: ACM, pp. 1419-1422.
- [8] Hirano, S.H. et al., 2010. vSked: evaluation of a system to support classroom activities for children with autism. In *Proceedings of the 28th international conference on Human factors in computing systems*. Atlanta, Georgia, USA: ACM, pp. 1633-1642. Available at: [file:///home/robert\\_peter/Work/Study/ResearchProject/p1633-hirano.pdf](file:///home/robert_peter/Work/Study/ResearchProject/p1633-hirano.pdf).
- [9] Kershaw, P., 2011. *The Asd Workbook Understanding Your Autism Spectrum Disorder.*, Jessica Kingsley Pub.
- [10] Konstantinidis, E.I. et al., 2009. Using affective avatars and rich multimedia content for education of children with autism. In *Proceedings of the 2nd International Conference on Pervasive Technologies Related to Assistive Environments*. Corfu, Greece: ACM, pp. 1-6.
- [11] van der Kooij, Martijn, PECS for All. Available at: <http://pecsforall.com/> [Accessed April 27, 2011].
- [12] MedicalMine Inc, ChARM Tracker. *Cloud-based affordable healthcare solutions*. Available at: <https://charmtracker.com> [Accessed June 11, 2011].
- [13] Milne, M., Powers, D. & Leibbrandt, R., 2009. Development of a software-based social tutor for children with autism spectrum disorders. In *Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7*. Melbourne, Australia: ACM, pp. 265-268.
- [14] Narayan, S. & Heiden, C., 2010. The cloud at work. *Crossroads*, 16(3), pp.19-22.



- [15] National Institute of Mental Health, 2004. Autism Spectrum Disorders (Pervasive Developmental Disorders). Available at: <http://nimh.nih.gov/health/publications/autism/nimhautismspectrum.pdf> [Accessed April 27, 2011].
- [16] Newschaffer, C.J. et al., 2007. The epidemiology of autism spectrum disorders. *Annual Review of Public Health*, 28, pp.235-258.
- [17] Øhrstrøm, P., HANDS Project - Home. Available at: <http://hands-project.eu/> [Accessed June 8, 2011].
- [18] Putnam, C. & Chong, L., 2008. Software and technologies designed for people with autism: what do users want? In *Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*. Halifax, Nova Scotia, Canada: ACM, pp. 3-10.
- [19] Quinn, B., 2011. *Autism spectrum disorders□: an altered perspective* 2nd ed., London: Jessica Kingsley Publishers.
- [20] Radicati, S. ed., Email Statistics Report, 2009-2013 Executive Summary. Available at: <http://www.radicati.com/wp/wp-content/uploads/2009/05/email-stats-report-exec-summary.pdf> [Accessed November 15, 2010].
- [21] Ranfelt, A.M., Wigram, T. & Øhrstrøm, P., 2009. Towards a handy interactive persuasive diary for teenagers with a diagnosis of autism. In *Proceedings of the 4th International Conference on Persuasive Technology*. Claremont, California: ACM, pp. 1-9.
- [22] Sampath, H., Sivaswamy, J. & Indurkha, B., 2010. Assistive systems for children with dyslexia and autism. *SIGACCESS Access. Comput.*, (96), pp.32-36.
- [23] Sanderson, D., 2009. *Programming Google App Engine: Build and Run Scalable Web Apps on Google's Infrastructure* 1st ed., O'Reilly Media, Inc.
- [24] Schmidt, C. & Schmidt, M., 2008. Three-dimensional virtual learning environments for mediating social skills acquisition among individuals with autism spectrum disorders. In *Proceedings of the 7th international conference on Interaction design and children*. Chicago, Illinois: ACM, pp. 85-88.
- [25] Sclera NPO, Sclera picto's. Available at: <http://www.sclera.be/index.php?taal=ENG> [Accessed July 3, 2011].
- [26] Velte, T., Velte, A. & Elsenpeter, R., 2010. *Cloud Computing, A Practical Approach*, McGraw-Hill, Inc.
- [27] Wikipedia authors, Autism therapies - Wikipedia, the free encyclopedia. Available at: [http://en.wikipedia.org/wiki/Autism\\_therapies](http://en.wikipedia.org/wiki/Autism_therapies) [Accessed April 27, 2011a].
- [28] Wikipedia authors, Picture Exchange Communication System - Wikipedia, the free encyclopedia. Available at: [http://en.wikipedia.org/wiki/Picture\\_Exchange\\_Communication\\_System](http://en.wikipedia.org/wiki/Picture_Exchange_Communication_System) [Accessed April 27, 2011b].
- [29] Yeganyan, M.T. et al., 2010. vSked: an interactive visual schedule system for use in classrooms for children with autism. In *Proceedings of the 9th International Conference on Interaction Design and Children*. Barcelona, Spain: ACM, pp. 319-322. Available at: [file:///home/robert\\_peter/Work/Study/ResearchProject/p319-yeganyan.pdf](file:///home/robert_peter/Work/Study/ResearchProject/p319-yeganyan.pdf).

# Appendices

1. **Software Products and User Experience** – printed version of the online survey
2. **Survey Evaluation** – a detailed evaluation of the survey responses
3. **CD** – Eclipse project library with source code