DESIGN OF AN EDUCATIONAL APPLICATION FOR APPLIED BEHAVIOR ANALYSIS AND DISCRETE TRIAL TRAINING

Barbara Zagajšek, Željka Car
Faculty of Electrical Engineering and Computing
University of Zagreb
Zagreb, Croatia
email: barbara.zagajsek@gmail.com, zeljka.car@fer.hr

Jasmina Frey Škrinjar
Faculty of Education and Rehabilitation Sciences
University of Zagreb
Zagreb, Croatia
email: frey@erf.hr

ABSTRACT
This paper presents requirements and design of an educational mobile application based on augmentative and alternative communication and aimed for persons with complex communication needs. The application prototype has been evaluated during educational sessions with children with autism spectrum disorders. The educational objective is learning new words based on Applied Behavior Analysis and Discrete Trial Training educational methods. The paper describes and compares the initial application design and its adaptation for integration within a software platform for augmentative and alternative communication.

KEY WORDS
Software design, Web services, Applied Behavior Analysis, Discrete Trial Training.

1 Introduction

Complex communication needs (CCN) occur when individuals cannot meet their daily communication needs through usual communication methods. They result from a variety of physical, sensory, and environmental causes which restrict independent functioning in society [1]. Persons with complex communication needs are not able to communicate temporarily or permanently by using speech and have difficulties in production and/or understanding of oral and/or written language. They could be of different ages and abilities, making this a highly diverse population in every sense. Some of them use alternative and augmentative communication (AAC) to compensate or reduce existing communication difficulties.

Autism Spectrum Disorders (ASD) are a group of neural development disorders of the brain with a so far unknown cause. ASD features are: significant disorder in social interaction, in communication (verbal and nonverbal) and behavioral difficulties such as restricted interests, activities and stereotypes [2]. These symptoms manifest themselves in children before the age of three. The term ‘spectrum’ refers to the fact that these disorders affect each individual in a different way with different degrees of severity [3]. Persons with ASD are persons with complex communication needs.

The increased number of educational ICT solutions designed for use on small mobile devices, particularly on devices with a touch screen, smartphones and tablet devices, add a new paradigm of using and distributing mobile applications. New and innovative approaches of learning for people with CCN have emerged. Due to high diversity of this user group, AAC based ICT educational applications should include options for efficient adaptation of content and user interfaces to a particular user’s needs and capabilities. Adaptation of educational content and presentation is time-exhausting and generates a significant amount of data on the device.

Work presented in this paper is part of a multidisciplinary research in the area of alternative and augmentative communication in Croatia 1,2. This research aims at finding efficient solutions for an integrated and holistic approach for accessing different communication and education services for different kinds of mobile users working on different devices in a mobile telecommunication network. Scientists and specialists from electrical engineering and computing, educational-rehabilitation sciences, psychology and graphic arts are involved in the research. Within this research an AAC software platform for component based development and deployment of inter-operable and scalable communication and education services is proposed [4] and implemented.

Analysis of the AAC user needs have shown, among others, that there exists a significant potential and need for implementing an application for learning new words based on Applied Behavior Analysis (ABA) and Discrete Trial Training (DTT) educational methods. This paper describes two versions of such an application, called aLearnWords. A native mobile version of the aLearnWords application for Android OS has been developed as a proof-of-concept prototype for implementation of ABA and DTT methods on mobile tablet devices. It has been evaluated in a rehabilitation center while working with Croatian preschool children with ASD. The initial design of the native application has taken into account its future integration within the AAC software platform in a way that is presented in

1Project "ICT systems for persons with complex communication needs" funded by University of Zagreb Development Fund January 2012-January 2013 (www.ict-aac.hr/svuclisti/en)
2Project "ICT competence network ICT Competence Network for Innovative Services for Persons with Complex Communication Needs" www.ict-aac.hr, funded by EU and launched in March 2013.
the paper and represents the main contribution of paper. Also, the paper describes the overall concept and necessity of creating an AAC platform and compares two potential software design approaches for developing this particular application, as well as any other AAC based educational application aimed for mobile devices.

The paper is organized as follows. Chapter 2 describes teaching methodologies for persons with complex communication needs and lists existing educational software solutions. In chapter 3 the aLearnWords application for teaching new words to CCN persons is presented in functionality and architecture. Chapter 4 describes the integration of aLearnWords within a software platform for augmentative and alternative communication. Finally, chapter 5 presents the conclusion.

2 Research Motivation - Teaching Persons with Complex Communication Needs

2.1 CCN persons and augmentative and alternative communication

Complex communication needs persons cannot communicate through usual communication methods due to various reasons that limit their independent participation in society. This condition can be temporary or permanent and this population is extremely diverse [1]. Some examples of conditions that can lead to complex communication needs are autism spectrum disorders, Down syndrome, cerebral palsy and various intellectual and motoric disabilities. Augmentative and alternative communication (AAC) helps people with CCN overcome the difficulties and challenges they face in everyday communication and education. It is an umbrella term encompassing all methods of communication for people that have difficulties in reproducing or understanding oral and/or written language [5]. The use of AAC can significantly improve CCN persons’ quality of life, enable them equal participation in society and better education as well as a higher degree of independence [6]. The extremely diverse nature of this population, as well as the high number of people suffering from various disabilities leading to complex communication needs, stresses out the importance of designing software that can be adjusted to a particular user to the maximum extent.

2.2 Applied behavior analysis and discrete trial training

Applied Behavior Analysis (ABA) is the science in which procedures derived from the principles of behavior are systematically applied to improve socially significant behavior to a meaningful degree and to demonstrate experimentally that the procedures employed were responsible for the improvement in behavior [7]. It represents a scientifically verified behavioral approach to teaching CCN persons with various degrees of disability [8]. Discrete Trial Training (Discrete Trial Instruction) (DTT) is a structured ABA technique used in early intervention for children with CCN, developed by a clinical psychologist Ole Ivar Lovaas [9]. DTT is a very efficient method in teaching specific skills: it is consistent, the goal of each task is clear and progress is measurable. DTT teaching comprises of a series of trials. Every trial is a teaching unit that consists of 3 components:

- Instruction -> behavior -> consequence

and is implemented by the tutor who works one on one with a child [10]. In each trial, an instruction is given to a child by the tutor. The child responds with a certain behavior. In case of a correct answer, the tutor will, as a consequence, give a reinforcer: a stimulus that follows a behavior and leads to increase in responding [7]. Reinforcers are basically rewards, they should be attractive to a particular child and serve as motivation for further cooperation and learning. Prompts are additional stimuli used to facilitate a correct response [7]. They help the child in recognizing the correct symbol (graphical representation of a new item, concepts or objects that the child is learning), but should be diminished as the child becomes more and more familiar with the symbol. Examples of prompts are physical guidance, demonstration, verbal directions and gestures. DTT is the current best practice in teaching children with CCN and represents one of many Applied Behavior Analysis strategies. It is also the teaching methodology implemented in the case study described later in this paper.

2.3 Existing educational software solutions for CCN persons

During the last couple of years there has been an increase in the number of AAC and educational applications for CCN persons, especially for the mobile market (such as iTunes Apple Store [11] and Google Play [12]). A few notable examples are: ABCD SW: Autistic Behavior & Computer-based Didactic Software [13], The autism project [14, 15], Project CLES [16], Abaris system [17], Discrete trial trainer [18], ABA1Program [19], ABA Basic [20], Behavior Tracker Pro [21], Dr. Brown’s Apps [22]. Although this increase in attention shows a positive general trend towards full integration of this population into society, these applications have mostly predefined content and difficulty levels and only some offer any kind of parametrization functionality. Due to the diverse nature and ability levels of persons with CCN, there exists a need for creating adaptable educational software solutions the user could customize to his/her liking. There is also the problem of many independent applications. Even if some personalization/customization functionality were implemented, it would have to be done over again with each new application - or again when migrating to a different client device (e.g. from an Android tablet to an iPad) and reinstalling applications. This problem would be solved by creating a platform for augmentative and alternative communication
and education, one that would contain all user data and serve as a basis for the development of new software solutions for this user group.

3 aLearnWords - An Android Application for Teaching New Words to Special Education Needs Persons

3.1 Application description

aLearnWords is an Android touch screen educational application meant for persons (primarily children) with special education needs developed as part of previously mentioned research. The teaching algorithm implemented in the application follows the principles of Applied Behavior Analysis and Discrete Trial Training. Its goal is to teach receptive labeling, which means that, after hearing an instruction (e.g. 'show me the flower'), the child should touch the corresponding symbol on screen.

The application was developed as a proof-of-concept prototype in order to evaluate the applicability of use of tablet devices in ABA DTT educational sessions for persons with special education needs at the center for rehabilitation in Zagreb (The Faculty of Education and Rehabilitation Sciences, University of Zagreb). Later chapters in the paper describe a platform for augmentative and alternative communication and education within which this application was later incorporated. This platform was at a concept stage at the aLearnWords prototype development phase so it was decided to implement and evaluate the prototype as a native Android application and only incorporate into the platform once it reached a stable version and should the evaluation show its applicability in everyday education and rehabilitation. The evaluation showed great potential and the application was received with great enthusiasm by tutors and users alike. The evaluation itself is out of this paper’s scope, but was a relevant factor in continuing with the AAC platform integration. The early testing and later evaluation of the native version made it easier to quickly accommodate any necessary changes to the application itself. The symbols used in the application are from the ARASAAC symbol gallery [23] and are specifically designed for use in augmentative and alternative communication. The application assumes a tutor is present during the educational session. The tutor’s responsibility is mainly to choose which category and which specific words will be taught that session, start the session for a specific child and give instructions to the child.

An educational session consists of a number of trials in which the child is given a certain number of symbols (1, 2 or 3) and should touch the correct one. The other symbols in the trial are called discrimination symbols. After a response from the child (correct, incorrect or timeout), the application gives appropriate feedback (reinforcement or reward) and moves on to the next trial, following the internal ABA DTT education logic. An example of a trial is shown in figure 1. The application defines the goal symbol for every trial, shows its label in the upper-right corner of the screen (e.g. ‘FLOWER’) and the tutor’s job is to read it in the form of an instruction.

![Figure 1. aLearnWords trial example](image)

The internal ABA DTT teaching logic defines the number of symbols shown to the user as well as a certain level of prompt, depending on a particular symbol’s level of adoption (i.e. status). The number of symbols corresponds to the symbol education phase, according to the ABA teaching methodology:

- **Isolation** 1 symbol (the correct one)
- **Discrimination** 2 symbols (the correct one + 1 discrimination symbol) shown in random order
- **Random rotation** 3 symbols (the correct one + 2 discrimination symbols) shown in random order

The aLearnWords application supports 3 (+1) levels of prompt: strongest, medium and weakest (+ no prompt) shown most-to-least and these are manifested by fading the discrimination symbols in the trial. Figure 2 shows an example of a trial in which the target symbol (in this case a flower) is in discrimination phase (1 discrimination symbol is shown along with the correct one) with strongest prompt (the symbol for the swing is extremely faded, indicating that this is not the correct symbol). Every educational session starts with a specific testing phase in which the user’s knowledge of selected educational content is tested. This is necessary because the child might have learned a new symbol sometime after the previous session (with that symbol), or may have forgotten one. This phase adjusts the status of each symbol used in that particular educational session. When the user successfully recognizes a symbol in random rotation with no prompt (the last and most difficult status), it transitions to a so called generalization phase and is considered adopted (learned). It will no longer appear (nor can be selected) as a target symbol during education, but may appear as a discrimination symbol. All user responses are saved and used for data processing: determining the symbol (educational content) status, determining symbols for the next trial, dynamic adaptation of certain teaching parameters etc.
aLearnWords also comes with a set of administrative and overview screens meant for use and later session analysis by the tutor(s). This includes the following:

- an overview of users, user categories and corresponding educational content along with
  - each educational content status
  - past trial details for each educational content
  - an option for including or excluding a particular educational content in the educational session (shown in figure 3)
- an overview of past educational sessions and trials for each user
- an overview and option to adjust teaching parameters for a particular user
- dynamic content adaptation log showing all parameter adjustments and trials that lead to a particular adjustment for a particular user

![Figure 2. aLearnWords trial example - discrimination with strongest prompt level](image)

These options can be accessed through a context menu in the main application screen or during an educational session using the options displayed in the top right corner of the screen (can be seen in figures 1 and 2). There is also a pause trial option (top left corner of the screen), necessary because all user responses are timed and also because there is a predefined timeout before moving on to the next trial if the user does not give any response.

The aLearnWords application is developed for touchscreen devices, facilitating an immediate interaction between the user and system. This is especially important for those people with special education needs who have trouble in grasping the relationship between the mouse (or other input device) and the screen cursor [24]. The application design is purposely simplistic so that the child can concentrate on learning the symbols, and not be distracted by loud colors or complex animations.

3.2 aLearnWords application architecture

aLearnWords is a standalone Android application that uses an internal SQLite database on the device for data storage. A systematic approach to the design and development of aLearnWords, a new service for a very specific and diverse user group, was applied. Insights from various specialists in the field of rehabilitation of persons with complex communication needs were used in a series of requirements elicitation iterations. Considering the domain, this process is complex and time-exhaustive. It is especially difficult to evaluate any solution on a significant number of end-users since their needs and abilities differ, sometimes to the extreme. As a solution to this complex situation, a development model with evolutionary prototype was applied (instead of throw-away prototyping) in the development of this native version of the application and its faster implementation within the described software platform. By using the developer’s knowledge and experience in the application of good design practices, a scalable and maintainable solution was developed, one that would later be easily integrated within a platform for augmentative and alternative communication and education, and one that would serve a very numerous and somewhat neglected group of users.

During application design, special considerations were made to clearly separate the presentation from the business logic and database access layer by programming to interfaces (supertypes) instead of particular implementations and encapsulating code that might change more frequently [25]. By making use of well know design principles and patterns, one can later easily switch to different implementations (a different database or domain logic algorithm). This is always a good approach, since requirements are frequently subject to change. The architecture of aLearnWords is shown in figure 4.

SQLiteDynamicABARepository represents the data access layer where all database queries, inserts and updates are implemented. In this case, the database in question is the internal SQLite database. This class implements the IDynamicABADataRepository interface which defines 28 (public) methods. The database schema is created and initial application data is inserted into the client database upon application installation. This includes a predefined set of users, categories and corresponding educational content

\[\text{Model designed with yEd Graph Editor v3.11 (©2000-2013 yWorks-GmbH), www.yWorks.com}\]
with initial statuses, and also predefined default teaching module settings for each user.

![Diagram](image)

Figure 4. aLearnWords application architecture

The ABA DTT domain logic is where all the domain classes and internal logic and algorithms reside. The LocalABAService service layer receives a request from the presentation, fetches, inserts or updates data if necessary by invoking methods on the data access layer and passes on required data to the ABA DTT domain logic. The domain logic then processes the data and returns the requested information such as data for the next trial or whether an educational content status should be updated or which teaching parameter should be adjusted. LocalABAService implements the IABAService interface which defines 14 methods.

The presentation layer’s only concern is reporting user choices and results and fetching relevant data via the LocalABAService and presenting it in a meaningful way. The presentation does not concern itself with the internal ABA DTT teaching logic, or the underlining database where data is stored. For example, these are the actions taken by the presentation layer during an educational session (the example is simplified):

1. The presentation layer needs data for the next trial. It invokes a method called getNextTrial (from LocalABAService) and passes the id of the current session

2. The service fetches relevant data for the current session, passes it on to the ABA DTT domain logic which processes the data and returns an ordered list of symbols the presentation layer should display to the user, along with the information which symbol is the target symbol. e.g. tree, fork, flower (is target), and which level of prompt to apply

3. The presentation layer displays the symbols to the user (applies prompts if necessary) and waits for user response (or a predefined timeout of 20 seconds)

4. The user responds (correct, incorrect, timeout) and the presentation layer invokes the method reportTrialResult and passes the result of the current trial, along with the reaction time and any other relevant data. The service layer now invokes necessary methods on the domain logic for data analysis and the data access layer for data updates.

5. The presentation layer continues with the next trial by repeating the steps listed above

Like the presentation layer, the domain logic or data access layer do not concern themselves with presentation issues such as symbol layout or prompt manifestation.

### 3.3 Evolutionary prototype in rehabilitation

Six sessions were conducted in order to test the final evolutionary prototype involving two educational rehabilitators and three children - boys, 3 to 5 years old, that are included in a special preschool group within regular kindergarten. The rehabilitators have more than 8 years of experience in teaching children with ASD. They conducted DTT in teaching using traditional materials (pictures, objects) and had a brief experience in using the aLearnWords application before this prototype testing. Since all children involved are nonverbal and have below average intellectual functioning, receptive labeling was the goal of their Individualized Education Programs and was taught using pictures and objects.

During prototype testing different data was collected. Among other, the data was analyzed in order to see the relevance of fluency for grading the value of correct or incorrect answers. 89% of answers were made within 4 seconds, so the fluency was really good. The interpretation of that result can be that the children were very motivated during teaching sessions and oriented towards the tablet with the installed aLearnWords application and using it. Since educational rehabilitators did other tasks that are not connected with this application, different behavior of a child when using it could be observed. When doing other tasks there were more disruptive behaviors (getting out of chair, turning head, throwing materials) and when working with the application on the mobile device children were sitting and orienting towards the application and the educational rehabilitator. This was also the case in the Moore and Calvert’s (2000) study [26] in which the authors found that children with autism were more attentive, more motivated, and learned more vocabulary while using a computer program rather than during traditional education.
4 Integrating aLearnWords within a Software Platform for Augmentative and Alternative Communication and Education

4.1 Motivation

Today, many software solutions for augmentative and alternative communication and education can be found, either as software installations for personal computers, Web or mobile applications. These are mostly self-contained applications that offer little or no customization or personalization features. Personalization usually implies adding a custom set of symbols for communication/education, but it can mean a series of other things such as defining the symbol order, position, presentation or size in the application. In this case, any personalization the user would make would be applied only on the device the application is currently installed on, causing problems for the user if he/she wanted to make a new installation or switch to a new device (a different touch screen tablet, for example) and keep all of his/her settings and data.

By creating a platform for AAC and educational applications, where all user data (settings, personalized data) would be stored in a server-side database, the problems listed above would be addressed. Multiple clients could be developed, irrespective of their native operating systems, since all server data would be accessed via Web services. This would mean shorter development time for most new features since there would be a set of implemented and already tested components such as user login, fetch symbol or educational content etc. The platform would serve as a centralized data and Web service repository upon which all future (different) AAC and educational applications would be built, along with their specific domain logic and data. This kind of distributed architecture would be very beneficial to the user since he/she would no longer be dependent on a particular computer or device, and would always have his/her personalized data with every application developed as part of the platform.

This novel approach in developing new, personalized and completely adaptable educational solutions for CCN persons as part of an AAC platform opens up a world of possibilities. The platform serves to unite (future) mobile OS and Web versions of the application that should be developed with extreme usability and suited for the intended user-group. For these users, usability, adaptability and the presence of AAC symbols they know and recognize is key.

4.2 Platform architecture

The software platform for augmentative and alternative communication and education is designed as a RESTful (REpresentational State Transfer) client-server architecture [27] with a RESTful Web Application Programming Interface (API) or RESTful Web service. RESTful Web services are a more light-weight, resource-oriented alternative to SOAP-based Web services [28]. The client-server communication is implemented through HTTP (Hypertext Transfer Protocol) requests (a specific URL for a specific resource on the server) and JSON (JavaScript Object Notation) [29] responses in the following way:

1. The client sends an HTTP (GET/POST/DELETE) request to the server
2. The server processes the request (fetching or updating data in the database if needed)
3. The server returns a JSON response to the client
4. The client processes the response and generates an appropriate view (presentation) to the user

The server-side is implemented in the Java programming language, using the Spring MVC framework [28, 30], Hibernate for relational database persistence [31] and a MySQL database [32]. The system architecture and server-side implementation are described in more detail in [33].

4.3 aLearnWords case study

Since the aLearnWords application was designed with separation of concerns in mind from the very start, it was fairly easy to replace specific implementations in necessary layers. The communication between the Android application and the server was implemented as was described earlier in chapter 4.2. The database schema is the same as in the standalone application, but migrated to MySQL database. This means it was necessary to create a new implementation of the data access layer. The new aLearnWords architecture as part of the platform for augmentative and alternative communication and education is shown in figure 5. The figure only covers the educational aLearnWords section of the platform. ABA DTT domain logic is the same as in the standalone application version, the only difference is that the logic now resides on a remote server. The data access layer has been replaced with a new implementation due to the change in the underlining database instance.

The Web service methods are implemented in controllers. All client requests are received by the Spring dispatcher servlet (front controller), and forwarded to appropriate controller methods based on the URL address. The controllers themselves perform little or no processing, they delegate the responsibility for the business logic to ABAServiceImpl which, in turn, communicates with the data access layer and ABA DTT domain logic. Data conversion to JSON objects on the server-side is handled by the Spring view resolver [28]. An extra adapter layer had to be added in the client application (aLearnWords) for adapting JSON objects received from the server to appropriate Java objects (POjos) the client application works with (not shown in the figure as the clients are simplified). Besides this, the clients themselves are left only with the task of presenting the data.
Initial clinical evaluation done by tutors having educational sessions with children with ASD have shown a significant potential of using the aLearnWords application.

By developing new software solutions for AAC and education of persons with CCN as part of the platform as earlier described, the end-user is no longer tied down to a particular client device and can easily switch to a different one, with a different OS (as long as there exists a client application developed for that OS). Every change to the user’s personalized data, preferences or settings is saved in a remote database and is therefore accessible to every other client application developed as part of the platform for AAC and education.

The software developer developing a new software solution for AAC or education can start making visible progress very early on in the software development cycle. He/she does not have to deal with designing and implementing the system architecture or framework since it already exists. He/she also has a ready database which can be expanded with custom tables, if necessary. Already implemented and tested frequently used components such as user login/logout, fetch user data, fetch symbols etc. contribute to an even quicker development time.

One negative aspect of this distributed architecture is the necessity of constant Internet connection which can be avoided through the development of synchronization mechanisms on each client. The development of new features specific to a particular application that require communication with the platform server will be somewhat more difficult due to the necessary communication via Web services and the adaptation of the server’s JSON responses to appropriate client-specific domain objects. Also, more effort would have to be made by the software developer when implementing a new component/Web service since this kind of development requires special considerations.

Scientists and specialists from electrical engineering and computing, educational-rehabilitation sciences, psychology and graphic arts have contributed to the conceptual idea and design of the described educational application. The application is a result of a synthesis of knowledge on persons with CCN (their thoughts, ways of learning and social cognition), information-communication technology and modern standards of visual graphic art design. Our goal was and is to contribute to innovative approaches of learning and quality of life of persons with CCN, their caregivers, but also the community in which the software tools developed enable a more inclusive, significant and easier participation for all. Further scientific research will show whether and to which extent our effort was successful. An obvious early benefit of this project are new insights and knowledge adopted by all researchers involved, which is a direct result of multidisciplinary collaboration. The experience in working on this project has given everyone involved a new dimension in reflecting on their own professional challenges.

5 Conclusion

References


