



# ELISA

## Fairy Shopping

### Exposee

The Elisa shopping list is a useful application that helps you buy products at the cheapest prices, at the nearest stores, in the perfect order and in cooperation with your friends and family! Never miss a special offer or forget to buy important products.

Elisa will take care of you.

Gernot Brunner, Stefan Cimander, Philipp Riedmann,  
Markus Schnappinger, Michael Schreier, Thomas Weber

## Vision

---

Shopping can be fun. We all know that. And shopping can be costly, nerve-wracking, time-intensive and annoying. And we all know that, too. We do not want this. We want to have fun.

But unfortunately it is usually the other way round. If you have to go shopping, there are two approaches how to handle that. The first possibility is to empty your mailbox every day, collect and study all leaflets and advertisements and make lists. Lists of special offers, lists of your daily needed supplies, lists of shops nearby or around your working place. After spending half of a day filling these lists, it is time to consider whether to buy butter at the nearest shop or drive two miles to a second shop in order to save 12 ct per piece because of special offers. After spending the second half of the day merging your lists and making plans where to buy what products and how to make the best of all the special offers, you are very proud of the result: A master plan that saves you a total of 5 euros. We all like the good feeling that we get by being so clever and saving money. But we do not like the thoughts of spending a whole day looking at special offers and planning.

The second possibility is to just enter a shop, buy products you consider attractive, get angry over the bill which is much higher than expected, being plagued by a guilty conscience all the way home because there clearly might have been healthier alternatives to the frozen pizza you just bought – just to experience that, once again, you forgot to reach for toilet paper. We do not like this scenario either.

Wouldn't it be great to have the good feeling of saving money without spending so much time? Wouldn't it be great to get a perfectly optimized masterplan in an effortless way? What a fairy experience that would be!

The Elisa shopping list app grants you this experience. All you have to do is to add items you want to buy to a list on your smartphone. Then the magic begins.

Elisa is performing all the nerve-wracking and time-intensive work for you. She traverses all special offers of your most frequented shops, looks up products in her huge database and automatically sorts your wishes following several optimization criteria. Whether you want to buy products at the cheapest prices, at the lowest time investment or stores of the shortest distance, or even a mix of all that - Elisa is there to do all the work for you. Elisa wants to make you feel good. Elisa does not want you to run out of toilet paper again. She remembers what products you buy regularly and reminds you to buy them again after a certain period of time. If there is a really cheap special offer of an article you might need, Elisa keeps you informed about the great opportunity.

As a female, Elisa is of course capable of multitasking. So don't be selfish, invite others to use Elisa, and you will experience her multi user features. Sharing and cooperative editing of lists is supported and an ideal opportunity for all household members to keep on track with the other persons' needs. You can share your whole list or just entries that are e.g. related to your next holiday. Elisa does not only make your life easier, but also your life with others.

## State of the Art

---

This chapter will focus on existing technologies mainly trying to optimize the planning part of shopping. In addition, an application is examined which provides information about special offers. Due to the scope of this document, a tabular representation using key words is chosen to present the existing approaches.

### **Wunderlist**

General to do list without product database or product categories

### **Einkaufsliste**

Auto completion, sharing of lists, product categories, prices and support of android watch  
Database maintenance by the user.

### **Out of Milk**

Shopping list featuring auto completion and shared lists, product categories and prices, barcode scanner  
Product categories and prices have to be administered by the user

### **Bring!**

Shopping list with focus on usability and design, but a lack of features

### **Paprika Rezept Manager**

Automatic creation of shopping lists based on imported recipes

### **Markt guru**

Provides pdf versions of special offer leaflets and search function for products

## Goals and Features

---

The main purpose of the Elisa application is to optimize both time investment and money spent during your weekly shopping. The complex application breaks down to three main aspects:

- A widespread database with up to date articles, product prices and availabilities has to be developed and maintained.
- Multi-user functionalities, authentication and data confidentiality issues have to be addressed.
- Special offers are identified and forwarded to the user due to his experienced habits.

This chapter illustrates the core functionality before further features are presented.

### **Core functionality**

Although all three above aspects are very import for Elisa, the unique selling point is identified in the optimization part. This feature is based on an intuitive shopping list, to which users can easily add entries. Afterwards, the optimization is initialized: Electronically available offers of all major super markets are used to determine the lowest price of every product. Based on that data, the initial shopping list can be divided into sub-lists, enabling the user to easily know what products to buy at which store. These sublists are generated not only based on price, but also with respect to the user's habits and a reasonable time invest. Further details for the configurability of the optimization criteria are presented in the next section.

Special Offers influence our shopping behavior to a great extent. But nowadays, in our fast moving and automated world, no one is interested in spending hours and hours scanning through special offers. Even those scrimpers who still do, would certainly appreciate the opportunity to outsource this work. Our application provides this opportunity. During the usage of the app, knowledge is generated about what products are bought very often or in very regular intervals – thinking of your daily cornflakes in the first case or durable articles like detergent in the second case. As soon as the program detects a special offer for your favorite cornflakes, the user is informed. The same procedure applies for commodities of the second type. The app suggests to refill the stock of the commodity because of the low price. So Elisa basically assumes the role of a loving grandma, who supports you to save money and never run out of toilet paper again.

## Additional features

The described core functionality provides a basis for further features. In order to enable people sharing a flat to use the application in its entirety, the app offers several multi user features. Lists can be shared and synchronized with peers, entries in a list can be accompanied by comments, and entries or sublists can be assigned to a certain user.

Often the price of a product is the most critical criterion, but sometimes also the invested time, the location of the store or organic images influence our shopping behavior. To model this, the user is able to configure the optimization algorithms by weighting the used criteria.

It is often useful to structure a list with many entries into smaller sub-lists. Such lists can be created for special events, for every contributing peers, or by the system.

User specific eating habits like allergies or diet plans can be entered. The system can show warnings if those guidelines are violated by an entry in the shopping list

Shopping behavior can be observed using smartphone sensors like GPS. The user's position can be approximated and the system can use that data to optimize the path in a super market. At the next time, the items are arranged on the list in a time saving order.

Recipes can be integrated into the application in order to enter the ingredients automatically to the shopping list.

Please keep in mind that these additional features are not part of the core functionality and are therefore uncertain to be implemented at full volume. For an overview about the actual state of the project please see correspondent chapter of this document.

## Challenges

---

Apart from the technical challenges, a big challenge is identified in the User Experience context. It is mandatory to configure Elisa in order to get best optimization results. List entries should be augmented with attributes concerning amount, category or purpose, multi-user data, a due date etc. Forcing the user to answer so many questions every time he enters a new item is not acceptable. Hence, the development process for the Elisa project is augmented with usability studies in order to meet the customer's needs to a great degree.

Technical challenges are identified in the contexts of data acquisition, frontend design, and learning of the application. Those are explained in the following, together with short sketches how we plan to approach them.

## Data Acquisition

As there are no APIs available to get data from super markets, it is a major challenge to collect both products and accompanying data like prices, special offers, package size etc. The data used in our backend is collected using automated web search with web crawlers. If this is not sufficient, image recognition can be applied to special offer leaflets.

## Frontend Design

Users shall use the app in an intuitive way and therefore find classical list features like drag and drop, completion of a shopping list entry, editing of an entry and so on. This functionality has to be augmented with special offer notifications, sublist features, and multi user functionality. It is a major challenge to keep a high degree of usability with so many features and options.

## Learning

Elisa offers a lot of configurability options. To limit the effort for users to a minimum, most settings can be learned automatically by the system. For example the order of the products within a store can be determined and used to order the list entries accordingly. Also specific preferences like 'what cornflakes I like most' can be learned. In addition, notification settings can be learned in order to avoid five notifications per day because of a detergent special offer.

## Project management

---

The Elisa project is a pushing-the-limits project hosted by the Software Engineering elite graduate program. Participating students are in alphabetical order Gernot Brunner, Stefan Cimander, Philipp Riedmann, Markus Schnappinger, Michael Schreier and Thomas Weber.

## Overall Process

Even though a student project, the Elisa project follows a defined process. There is a team for each frontend and backend with one participant in the managing role. After a major requirements engineering phase at the beginning of the project, the requirements were broken down into user stories.

Every week a sample of user stories is selected and prepared. These are then discussed, redefined, and prioritized by the whole group in a grooming session. Afterwards the two teams assign stories to participants for implementation. To merge the user story into the master repository, a review is mandatory.

## Schedule

The schedule will regularly be discussed with the academic advisor and this section will be updated afterwards.

Inception	June 2016	<b>Inspiration and Registration</b> <ul style="list-style-type: none"> <li>* Brainstorming</li> <li>* Mind-mapping</li> <li>* Identification of core features</li> <li>* Registration as pushing the limit project</li> </ul>
	October	<b>Project Start</b> <ul style="list-style-type: none"> <li>* Roles and Responsibilities</li> <li>* Definition of the development process</li> <li>* Used technologies</li> </ul>
Kickoff		
Elaboration		<b>Requirements Engineering</b> <ul style="list-style-type: none"> <li>* Defining of Use Cases</li> <li>* Defining of Proto-Personas</li> </ul>
		<b>Data Collection</b> <ul style="list-style-type: none"> <li>* Data acquisition with Webcrawlers</li> <li>* Database set up</li> </ul>
	November	<b>User Interface</b> <ul style="list-style-type: none"> <li>* Basic List functionalities</li> <li>* Frontend design following material design guidelines</li> </ul>
		<b>Data Transformation</b> <ul style="list-style-type: none"> <li>* Definition of an internal data structure</li> <li>* Data transformation script</li> </ul>
Elaboration Implementation		<b>Infrastructure Setup</b> <ul style="list-style-type: none"> <li>* Node-Server and NGINX on AWS server</li> <li>* Assumptions for internal API Design</li> </ul>
Implementation	December	<b>Working Prototype</b> <ul style="list-style-type: none"> <li>* Connection of Frontend and Backend</li> <li>* Log in and registration</li> <li>*</li> <li>* Yet to be defined</li> <li>*</li> </ul>
Refinement	January	<ul style="list-style-type: none"> <li>* Yet to be defined</li> </ul>
Delivery	February	<b>Delivery</b> <ul style="list-style-type: none"> <li>* Presentation</li> <li>* Documentation</li> </ul>

# State of the Project

---

## Working Prototype

The database is hosted on a server of the Amazon Web Services as well as Node and NGINX instances. An API is under construction.

## Database

Two web crawlers have collected several thousand datasets from the webpages of REWE and EDEKA. These data have been evaluated and transformed into our own storage format suitable for analysis and cross market comparison. A process to store user log in data is under elaboration.

## Frontend

Basic List functions are implemented such as entering an item to the list, editing, reordering via drag and drop, complete an item of the shopping list and redo of the completion. A log in form has been created. Lacking a connection to database at this state of the project, the frontend temporarily uses the key-value store of the browser. The frontend design is determined and will be applied from now on.