DOI: 10.15514/ISPRAS-2019-31(3)-2

Graphic DSL for Mobile Development

A. Gudiev, ORCID: 0000-0002-0674-8621 <arturgudiev93@gmail.com>
A. Grazhevskaya, ORCID: 0000-0002-5069-443X <sagrapro7@gmail.com>
Saint Petersburg State University,
7/9 University Embankment, 199034, Russia.

Abstract. Due to the increase in the number of platforms, languages and methods which are used in mobile development, the general technology elaboration problem is quite relevant nowadays. Graphic languages simplify software development, allowing to present program structure in terms of visual diagrams. Besides, graphic languages allow software engineers to avoid a lot of mistakes at the initial stages of design and development. Graphic domain-specific languages (DSL) facilitate application development by use of concrete domain abstractions. In this approach the mobile application structure will be presented in the form of various controllers connected among themselves through ports and corresponding to some complete fragments of logic. Controllers in turn consist of various states which allow to describe a data flow in the controller using various element connections. In each state the UI form which contains the graphic primitives and events connected with primitives can be described. Besides, code generator for UbiqMobile platform is implemented which will allow to generate UbiqMobile applications by the visual diagrams. At the end of the article demonstration examples on which the implemented DSL language was tested are given. The application allowing the user to get the trains schedule is provided in the first example. In the second application the user can log in to receive a check-in code.

Keywords: dsl; mobile development

For citation: Gudiev A.V., Grazhevskaya A.S. Graphic DSL for mobile development. Trudy ISP RAN/Proc. ISP RAS, vol. 31, issue 3, 2019. pp. 29-34. DOI: 10.15514/ISPRAS-2019-31(3)-2

Графический DSL для разработки мобильных приложений

А.В. Гудиев, ORCID: 0000-0002-0674-8621 <arturgudiev93@gmail.com>
А.С. Гражевская, ORCID: 0000-0002-5069-443X <sagrapro7@gmail.com>
Санкт-Петербургский Государственный Университет,
199034, Россия, г. Санкт-Петербург, Университетская набережная 7-9

Аннотация. В связи с увеличением количества платформ, языков и методов, использующихся в разработке мобильных приложений, задача выработки общей технологии довольно актуальна. Графические языки упрощают разработку, позволяя представить структуру программного обеспечения в виде графических диаграмм. Кроме того, графические языки помогают избежать множества ошибок еще на начальных этапах проектирования и разработки. Графические предметно-ориентированные языки (DSL) облегчают разработку программ путем применения абстракций конкретной предметной области. В данной работе представлен архитектурный шаблон мобильного приложения и созданный на его основе графический DSL, позволяющий описывать основную структуру мобильного приложения в терминах котнроллеров, состояний и переходов между ними. При таком подходе структура мобильного приложения будет представлена в виде различных контроллеров, связанных между собой при помощи портов и соответствующих некоторым целостным фрагментам логики. Сами контроллеры в свою очередь состоят из различных состояний, которые позволяют описать поток данных в контроллере путем соединения при помощи элемента-связи. В каждом состоянии может быть описана экранная форма, в которой содержатся графические примитивы и связанные с ними события, срабатывающие при их изменении. Кроме того, для разработанного DSL реализована автоматическую генерация кода для платформы UbiqMobile. В конце статьи приводятся демонстрационные примеры, на которых был апробирован DSL язык. В качестве первого примера приводится приложение, позволяющее пользователю посмотреть расписание электричек. Во втором приложении пользователь может войти в систему для того, чтобы получить check-in код.

Ключевые слова: предметно-ориентированные языки; мобильная разработка

Для цитирования: Гудиев А.В., Гражевская А.С. Графический DSL для разработки мобильных приложений. Труды ИСП РАН, том 31, вып. 3, 2019 г., стр. 29-34 (на английском языке). DOI: 10.15514/ISPRAS-2019-31(3)-2

1. Introduction

A large number of platforms, languages, and methods are used in mobile application development. Existing mobile development tools significantly differ from each other, and the common technology implementation problem is still relevant.

There are various ways of the high-level description of mobile application – architectural patterns mvc, pac, microkernel, etc [1]. All these patterns were borrowed from other software areas, are quite actively applied during mobile application development, but not quite correspond to their nature. Mobile applications differ from desktop and web programs [2]. Mobile applications are commonly used for short sessions, more focused on specific objectives performance.

Use of a suitable architectural pattern allows to increase considerably application development efficiency, but a bigger result can be achieved by graphic languages usage. DSL is the programming language in terms of the concrete subject domain which is applied to the solution of concrete type tasks [3], [4], [5]. Graphic DSL languages help to represent applications using visual diagrams. The result code will be generated according to these diagrams.

The purpose of this article is to develop an architectural template for mobile applications and to create graphic DSL based on it. DSL should allow describing the main logical application structure in terms of states, controllers and transition conditions between them.

2. Tools

The Modeling SDK technology is used for the graphic DSL implementation [6]. Modeling SDK is the plugin for Visual Studio intended for visual domain-specific languages development. Visual DSL development happens in the following order. At first, the metamodel (the set of all syntactically correct diagrams) is developed and edited, the implemented classes are generated. Then a DSL package compilation and debugging take place in an experimental instance of Visual Studio.

For metamodel programming, the graphic editor of Modeling SDK is used, but also it is possible to redefine or add new methods to the generated partial classes of the C\# language. The T4 language is used for code generation [7]. The Dsl and DslPackage projects are automatically created in the new solution of Visual Studio. In the Dsl project, various metamodel artifacts of the created DSL are stored. DslPackage project contains the user interface settings.

3. Controllers and states model

An application state corresponds to some complete logic fragment. The result of state change is data transfer which is logically finished and clear to other states.

It is convenient to group states and transition conditions into controllers by their logical connectivity, data community, UI forms, transition frequency and data transfer between states. Grouping states into controllers gives an opportunity to define more strong transition logic, allowing transitions between states in the controller and forbidding them between conditions of unconnected controllers.

The main application cycle is run by the special mechanism starting and switching controllers of states. Each controller has an entry point and an opportunity to set input parameters when an application switches to it. There can be several exits in a state. An application can return to the caused controller, switch to the next controller, etc. Execution logic is implemented in terms of the finite-state machine in the controller. Each controller has a set of the predefined states (in particular, initial and final states), and it is possible to add new states.

Mobile application implementation by means of controllers and states model allows to centralize its logical basis, the structure of the code becomes evident. When using controllers, the aim of the mobile application developer comes down to describing the necessary logical controllers, state and conditions of transition.

4. Graphic DSL Description

The model of controllers and states was tested on mobile applications of different classes and proved the efficiency. But the best results can be achieved, having taken this model as a basis of graphic DSL for mobile applications (see fig.1)

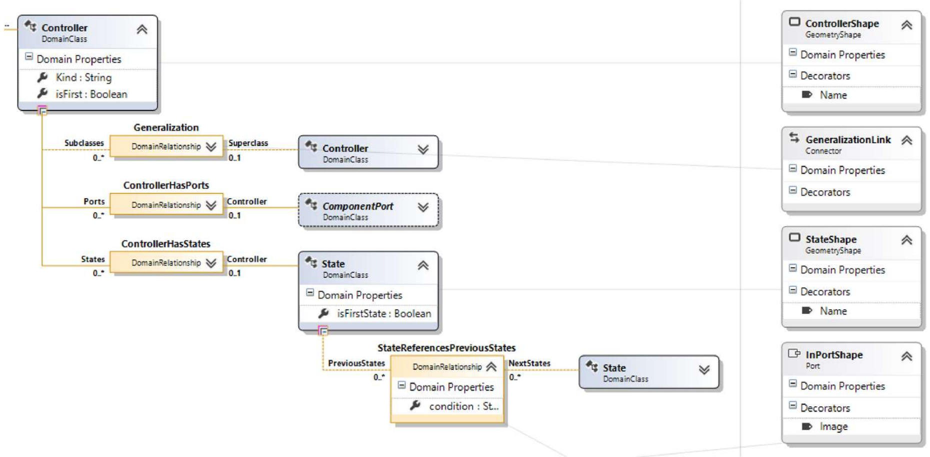


Fig. 1. Language implementation in Modelling SDK

Basic elements of the language are controllers and its states. States are placed on the controller, can connect among themselves and also to ports of the controller for the conditions description of an entrance and an exit from it. Each state opens in the separate diagram on which conditions of an entrance, an exit from a state and its internal logic are described. The logic of states includes a display of UI forms, processing of their events, services calls, conditions checking, etc. There is a display of a UI form for each state in the language. To connect the existing screen form with a state the ShowForm element is used.

5. Code generation

The language of T4 templates is used for code generation. The main components of the T4 language are directives, blocks of the text and control units. For a generation of the unchangeable code, text blocks are used, and dynamic parts are implemented by means of control units.

As a result of generation, the controllers' classes appear. Each controller has several states presented in the transfer type form. Process of work is implemented in terms of the finite-state machine. On links between states, the template of transitions are implemented. Controllers can also have ports. Ports are used for transitions between controllers.

The resulting code is applied to UbiqMobile platform [8]. UbiqMobile platform is aimed to cross-platform mobile development. The main features of the platform are that the business logic of all applications is executed on the server. And mobile devices have only thick clients to represent the result of application work.

6. Samples

The purpose of the first sample is to display the train schedule for the user. The application consists of a single controller and two states. In start state, the user can choose departure and destination stations (see fig.2). After clicking on the button, the application will switch the current state from the first state to the second one (see fig.3).

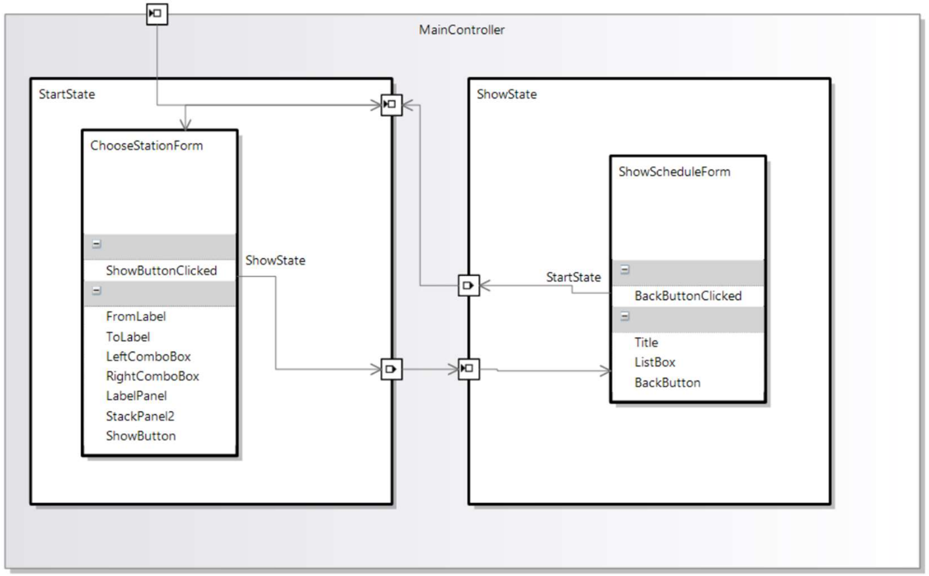
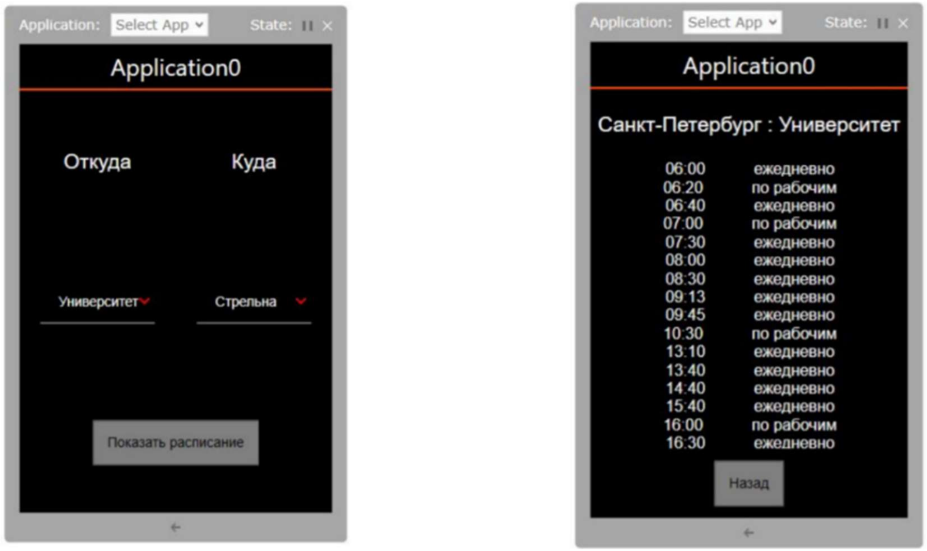


Fig. 2. Schedule application scheme



06:00	ежедневно
06:20	по рабочим
06:40	ежедневно
07:00	по рабочим
07:30	ежедневно
08:00	ежедневно
08:30	ежедневно
09:13	ежедневно
09:45	ежедневно
10:30	по рабочим
13:10	ежедневно
13:40	ежедневно
14:40	ежедневно
15:40	ежедневно
16:00	по рабочим
16:30	ежедневно

Fig. 3. Schedule application UI forms

The second sample allows the user to log in and receive the code which then can be used later (see fig. 4) There are two controllers in the application: LoginController and MainController. There is also a switching between controllers implemented by means of ports. In LoginController there is only one state. At MainController there are two states: a state with option selection and a state where a user can receive the necessary code. The UbiqMobile UI forms, corresponding to states of the application are given below (see fig. 5).

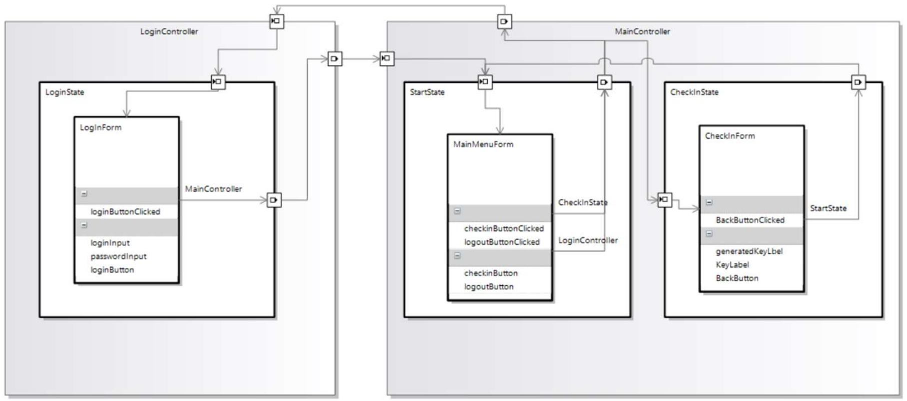
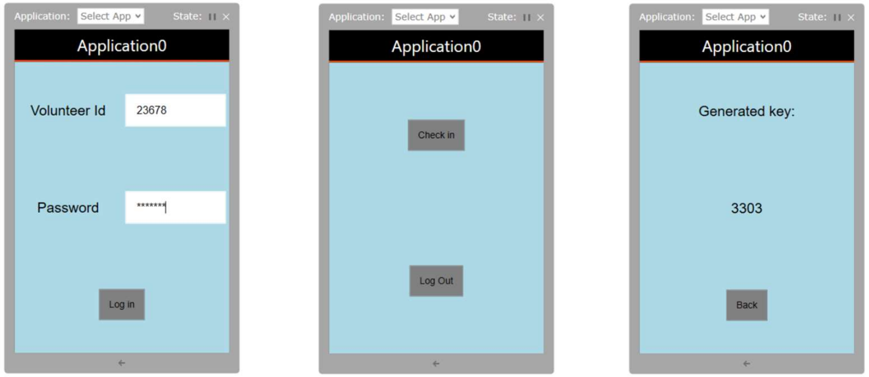


Fig. 4. Application with authorization scheme


Application: Select App State:
Application0
Log in
Application: Select App State:
Application0
Check in
Log Out
Application: Select App State:
Application0
Back

Volunteer Id 23678

Password ******

Generated key:

3303

Fig. 5. UbiqMobile screens

7. Conclusion

Within this work, the following results were achieved. The graphic DSL for mobile application development is implemented. The code generation for UbiqMobile platform feature is added. Demonstration samples are represented.

References / Список литературы

[1]. Plakalovic D and Simic D. Applying MVC and PAC patterns in mobile applications. arXiv preprint arXiv:1001.3489, 2010.

[2]. Flora Harleen K and Wang Xiaofeng and Chande Swati V. An investigation on the characteristics of mobile applications: A survey study. International Journal of Information Technology and Computer Science, vol. 6, issue 11, 2014, pp. 21-27.

[3]. Koznov D. Methodology and tools for domain-specific modeling. Thesis for the degree of Doctor of Technical Sciences, St. Petersburg State University, 2016 (in Russian) / Кознов Д.В. Методология и инструментарий предметно-ориентированного моделирования. Диссертация на соискание учёной степени доктора технических наук, СПбГУ, 2016.

[4]. Bryksin T.A. The platform for creation of specialized visual development environments of the software, PhD Thesis, St. Petersburg State University, 2016 (in Russian) / Брыксин Т.А. Платформа для создания специализированных визуальных сред разработки программного обеспечения. Диссертация на соискание учёной степени кандидата технических наук, СПбГУ, (2016).

[5]. Bryksin T.A. and Litvinov Yu.V. Environment of visual programming of QReal : Robots. In Proc. of the international conference on Information technologies in science and education, 2011, pp. 332-334 (in Russian) / Брыксин Т.А., Ю.В. Литвинов. Среда визуального программирования роботов QReal: Robots. Материалы международной конференции «Информационные технологии в образовании и науке», 2011, стр. 332-334.

[6]. Modeling SDK for Visual Studio – Domain-Specific Languages. Available at: https://docs.microsoft.com/ru-ru/visualstudio/modeling/modeling-sdk-for-visual-studio-domain-specific-languages, accessed 14.07.2019.

[7]. Code Generation and T4 Text Templates. Available at: https://docs.microsoft.com/ru-ru/visualstudio/modeling/code-generation-and-t4-text-templates?view=vs-2015, accessed 14.07.2019.

[8]. Onossovski V.V. and Terekhov A.N. Ubiq Mobile – a new universal platform for mobile online services. In Proc. of the 6th seminar of FRUCT Program, 2009, pp. 96-105.

Информация об авторах / Information about authors

Артур Владимирович ГУДИЕВ – магистр, выпускник кафедры системного программирования математико-механического факультета СПбГУ. Сфера научных интересов: мобильная разработка, предметно-ориентированное моделирование.

Artur Vladimirovich GUDIEV is a graduate of software engineering department of Mathematics and Mechanics faculty, St.Petersburg State University. Research interests: mobile development, domain-specific modelling.

Александра Сергеевна ГРАЖЕВСКАЯ – выпускница математико-механического факультета СПбГУ. Ее научные интересы включают предметно-ориентированные языки.

Alexandra Sergeevna Grazhevskaya is a graduate of the Mathematics and Mechanics faculty, St. Petersburg State University. Her research interests include domain-specific languages.