COMSATS University Islamabad

Abbottabad, Pakistan

SmartEditor

By

Hussain Ahmed **CIIT/FA19-BCS-074/ATD**

Ahmed Haroon **CIIT/FA19-BCS-018/ATD**

Zohaib Nadeem **CIIT/FA19-BCS-028/ATD**

Supervisor

Ms. Humaira Jabeen

Bachelor of Science in Computer Science (2019-2023)

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.

COMSATS University, Islamabad Pakistan

SmartEditor

A project presented to

COMSATS Institute of Information Technology, Islamabad

In partial fulfilment

of the requirement for the degree of

Bachelor of Science in Computer Science (2019-2023)

By

Hussain Ahmed **CIIT/FA19-BCS-074/ATD**

Ahmed Haroon **CIIT/FA19-BCS-018/ATD**

Zohaib Nadeem **CIIT/FA19-BCS-028/ATD**

DECLARATION

We at this moment declare that the software, including the report, titled "SmartEditor," has been developed entirely by us and is an original creation. We have not copied any part of the project from any source.

In case any portion of this project is found to be copied or reproduced from another source, we take full responsibility for the consequences. Furthermore, we affirm that no part of this work has been submitted to any other university or institute of learning for any degree or qualification.

We stand by the authenticity and originality of this project and are confident that it meets the highest standards of academic integrity.

Hussain Ahmed

Ahmed Haroon

Zohaib Nadeem

CERTIFICATE OF APPROVAL

This is to certify that the Year Project of BS (CS) titled "SmartEditor" was successfully developed by Hussain Ahmed (CIIT/FA19-BCS-074), Ahmed Haroon (CIIT/FA19-BCS-018), and Zohaib Nadeem (CIIT/FA19-BCS-028) under the guidance and supervision of Ms Humaira Jabeen.

In the opinion of the supervisor, the project is fully adequate, both in terms of its scope and quality, and meets the requirements for the award of the degree of Bachelor of Science in Computer Sciences.

Supervisor

External Examiner

Head of Department

(Department of Computer Science)

EXECUTIVE SUMMARY

The SmartEditor is a cutting-edge PDF editor program created to offer a complete answer for editing PDF files. Text and image editing, page manipulation, annotation, file conversion, printing, collaboration, security, compatibility, and a user-friendly interface are just a few of the many features it offers.

With SmartEditor, users can edit PDF documents without losing the original content, and convert them to editable Word documents. The app provides the flexibility and convenience of editing PDF documents on the go, without the need for specialized software or the original source file.

SmartEditor is an ideal tool for individuals and organizations looking to improve efficiency and productivity by making changes to PDF documents quickly and easily.

Users can open and edit PDF documents produced by other software thanks to the app's compatibility feature, which also ensures the security and integrity of the PDF document.

Overall, the SmartEditor is an essential tool for anyone who needs to edit and modify PDF documents, providing a convenient, intuitive, and efficient solution.

ACKNOWLEDGEMENT

We would like to begin by expressing our sincere gratitude to everyone who has been a part of our journey towards the successful completion of this project. We feel truly blessed and privileged to have had the opportunity to work on this challenging project, and we could not have achieved our goal without the support of many.

First and foremost, we would like to thank our project supervisor, Ms. Humaira Jabeen, for her exceptional guidance, support, and mentorship throughout the project. Her insights, feedback, and unwavering commitment to our growth and development have been instrumental in shaping our work and ensuring its success. We are truly grateful for her contribution to this project.

We would also like to extend our heartfelt thanks to our families and loved ones for their support, encouragement, and belief in us. Their unwavering faith, love, and motivation have been the driving force behind our dedication and commitment to this project. Their constant encouragement and support have inspired us to overcome challenges and reach our full potential.

Lastly, we would like to express our gratitude to our colleagues and peers who have provided us with their valuable insights, feedback, and assistance throughout this project. We are grateful for the opportunity to collaborate and learn from each other, and for the support, we have received from the wider community.

We are humbled by the support and encouragement we have received, and we take pride in the successful completion of this project. Once again, thank you to everyone who has contributed to this project, and we look forward to applying the skills and knowledge we have gained to future endeavours.

Hussain Ahmed

Zohaib Nadeem

Ahmed Haroon

ABBREVIATIONS

SRS	Software Require Specification
PC	Personal Computer
SDS	Software Design Specification
PDF	Portable Document Format
PNG	Portable Network Graphic

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. Introduction

The purpose of the Smart Editor app is to allow users to edit PDF documents by converting them to editable Word documents, without losing the original content of the PDF. The scope of the app includes editing text and images, adding and deleting pages, splitting and merging documents, annotation and markup, conversion, and security. The app is a valuable tool for individuals and organizations looking to edit and modify PDF documents, as it allows for changes to be made quickly and easily without the need for specialized software or the original source file. The design and implementation constraints of the app include limited screen size and resolution, limited processing power and memory, varying connectivity and network conditions, and the need to support a range of mobile devices and operating systems. The app's use case diagram includes opening PDFs, editing text, adding images, saving the document, and exporting to different formats. Overall, the Smart Editor app is designed to be a user-friendly, efficient, and reliable tool for editing and modifying PDF documents.

1.1. Brief

A PDF editor application is a software tool that allows users to create, edit, and modify PDF documents. Some of its key features include the ability to add, delete, or modify text and images within the document, the ability to add, delete, or rearrange pages, annotate the document with notes or highlights, export the document to other file formats, print the document, collaborate with multiple users to edit and annotate a PDF document simultaneously, ensure the security and integrity of the PDF document using password protection and digital signatures, and ensure compatibility with PDF documents created by other software. A PDF editor application can be a valuable tool for individuals and organizations looking to edit and modify PDF documents, and can help to improve efficiency and productivity by enabling users to make changes to documents quickly and easily.

1.2. Relevance to Course Modules

Object-Oriented Programming: In this course, we've studied the principles and practises for creating object-oriented software systems. Designing and putting into practise different objects that can manipulate PDF files would be the task of SmartEditor.

Different data structures and algorithms used in computer science are covered in this lesson on data structures and algorithms. Data structures like arrays, linked lists, stacks, queues, and trees have been covered in class. To keep and manage the data in the PDF files, we need to use some of these data structures.

Database Systems: We learned how to plan and execute database systems in this course. In our initiative, PDF file information, like metadata or annotations, will be stored in a database.

We learned about the concepts and methods used in user interface design through the study of human-computer interaction. A user-friendly interface that enables users to carry out various editing and manipulation duties on PDF files is necessary for your project, SmartEditor.

In this training, we have learned about the procedure for creating software systems, which is known as software engineering. To make SmartEditor reliable, maintainable, and scalable, the software engineering method would need to be followed.

1.3. Project Background

Users can easily manipulate and edit PDF files on their mobile devices thanks to the SmartEditor mobile programme. In many different sectors, PDF files are frequently used for documents like contracts, invoices, and reports. Without the proper tools, altering these files can be difficult, especially for users who are on the go.

The goal of the SmartEditor app is to give users a simple, effective method to edit and work with PDF files on their mobile devices. Users of this app can quickly edit, add, remove, or rearrange the pages in their PDF files, annotate, highlight text, and digitally sign documents.
Users of the application can also combine numerous PDF files into a single document or convert various file formats to PDF.

The user-friendly interface of the SmartEditor app will make it simple for users to navigate and finish tasks swiftly. By offering encryption and password protection, it will also give users a secure environment in which to work with confidential data, such as financial or legal papers.

1.4. Literature Review

In the current digital age, PDF editing and manipulation is a highly relevant and sought-after field. There are many current tools and technologies on the market that offer different functionalities for editing and manipulating PDF documents.

Adobe Acrobat, Foxit PhantomPDF, Nitro Pro, and PDFelement are a few examples of well-known PDF processors. These editors provide a variety of features, including the ability to add or remove pages, annotate text, underline content, and sign PDF files. The majority of these editors, though, are created for desktop and notebook use, so they might not be appropriate for portable electronics.
Mobile PDF tools have become increasingly popular in recent years. Mobile-based PDF editors with features comparable to desktop editors, such as PDF Expert, Xodo, and GoodReader, are available and make it simpler for users to edit PDF files while on the go.

There are ongoing research and development initiatives in the area of PDF editing and manipulation in addition to already existing products. By using image analysis methods, for instance, a recent study suggested a new method for improving the effectiveness of PDF document comparison. The research showed that, when compared to conventional methods, the proposed method detected differences between PDF documents with better accuracy and speed.

1.5. Analysis from Literature Review (in the context of your project)

Based on the literature review, it is evident that there is a growing demand for mobile-based PDF editors. The current mobile-based PDF editors offer several features, but there is still room for improvement in terms of usability, speed, and efficiency.

The SmartEditor mobile application aims to address these limitations by providing a user-friendly interface and efficient editing and manipulation functionalities for PDF files on mobile devices. It also focuses on security and privacy by providing encryption and password protection.

1.6. Methodology and Software Lifecycle for This Project

The methodology and software development life cycle (SDLC) model selected for the SmartEditor project is the Agile SDLC model. This model is well-suited for software development projects with changing requirements, as it emphasizes collaboration, flexibility, and continuous delivery.

1.6.1. Rationale behind Selected Methodology

For the SmartEditor project, the Agile SDLC model was chosen for a number of factors. First off, the project entails creating a mobile application, which calls for adaptability and flexibility to meet shifting needs. The Agile model offers an adaptable and iterative methodology that can take stakeholder input into account as requirements change and can handle this throughout the development process.

Second, the Agile approach lends itself well to team collaboration and communication, which are crucial to the project's success. The Agile approach places a strong emphasis on regular team meetings and communication with stakeholders, which can help to make sure that everyone is on the same page with the objectives and specifications of the project.

Thirdly, the Agile model places a strong emphasis on delivering functional software in shorter cycles, which can aid in ensuring that the project stays on track and that the final product satisfies the requirements of its users. This method can assist in identifying problems early on in the development process, lowering the possibility of delays and unforeseen expenses.

1.6.2. Rationale behind Selected Methodology

The SmartEditor project will adhere to Object-Oriented Programming (OOP) concepts when developing its software. OOP is a paradigm for computing that places an emphasis on using objects and their interactions to create software. The code may become more modular, reusable, and simpler to maintain as a result of this strategy.

2. Problem Definition

The problem that the SmartEditor project aims to solve is the limited availability of efficient and user-friendly mobile-based PDF editors. Existing mobile-based PDF editors often lack essential features, are cumbersome to use, and are not optimized for mobile devices, resulting in a frustrating user experience.

Furthermore, the widespread use of PDF files in various industries and sectors, such as education, healthcare, and finance, highlights the need for a reliable and secure PDF editor that can handle complex documents while maintaining user privacy.

2.1. Problem Statement

Despite the widespread use of PDF files in various industries and sectors, existing mobile-based PDF editors are often limited in their features and usability, resulting in a frustrating user experience. This creates a need for a reliable and efficient mobile-based PDF editor that is optimized for mobile devices and meets the needs of users in various industries and sectors. The SmartEditor project aims to address this problem by developing a user-friendly, efficient, and secure mobile-based PDF editor that allows users to easily edit and manipulate PDF files on their mobile devices, with a focus on streamlining the user experience and ensuring user privacy.

2.2. Deliverables and Development Requirements

Mobile application: The primary deliverable of the SmartEditor project is a mobile application that allows users to easily edit and manipulate PDF files on their mobile devices.

User interface design: A streamlined and intuitive user interface design is required to ensure that users can efficiently and effectively use the application to edit and manipulate PDF files.

PDF editing features: The application must provide essential PDF editing features, such as adding, deleting, and rearranging pages, highlighting and annotating text, and applying encryption and password protection to ensure user privacy.

Compatibility: The application must be compatible with various mobile devices and operating systems, including iOS and Android.

Security: The application must be designed with security in mind to ensure that user data and privacy are protected.

Testing: The application must undergo rigorous testing to ensure that it functions properly and that all features are working as intended.

Documentation: A comprehensive user manual must be created to help users understand how to use the application and its features.

Deployment: The application must be deployed to the appropriate app stores for users to download and use.

Maintenance: Ongoing maintenance and updates will be required to ensure that the application continues to function properly and remains compatible with new mobile devices and operating systems

Development requirements:

Development tools: A suitable integrated development environment (IDE) that supports software development for the target platform.

Programming languages: Programming languages such as Java, C++, or Python that are suitable for developing the software.

Frameworks: Frameworks and libraries for developing software with PDF manipulation capabilities.

Hardware requirements: Sufficient hardware resources such as processor speed, RAM, and storage to support the development environment and run the software.

Team members: A team of developers with suitable skills and experience in software development and PDF manipulation.

Project management: A suitable project management methodology to guide the development process and ensure timely delivery of the software.

2.3. Current System (if applicable to your project)

SmartEditor aims to provide an easy-to-use and efficient solution for editing and manipulating PDF documents. Its unique features include support for various PDF file formats, the ability to merge and split PDF files, annotate PDF documents, edit PDF content, and more. Additionally, the app is designed to provide a user-friendly interface and is optimized for performance.

Compared to other PDF editing and manipulating apps, SmartEditor offers a comprehensive set of features that make it stand out. Its intuitive design and smooth functionality provide users with a hassle-free experience while working with PDF documents. The app's high-level security protocols ensure that all user data is kept safe and secure.

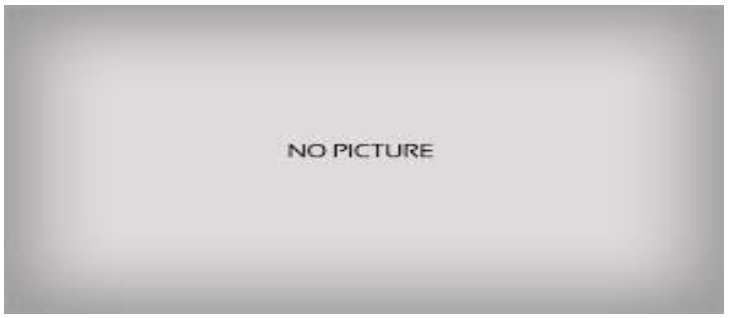


Figure 2.1: Sample picture

The following table (Table 2.1) is sample table; you are required to follow the same style of numbering and caption for the whole report.

Table 2.1: Sample Table

Header 1	Header 2	Header 3
Text	Text	Text

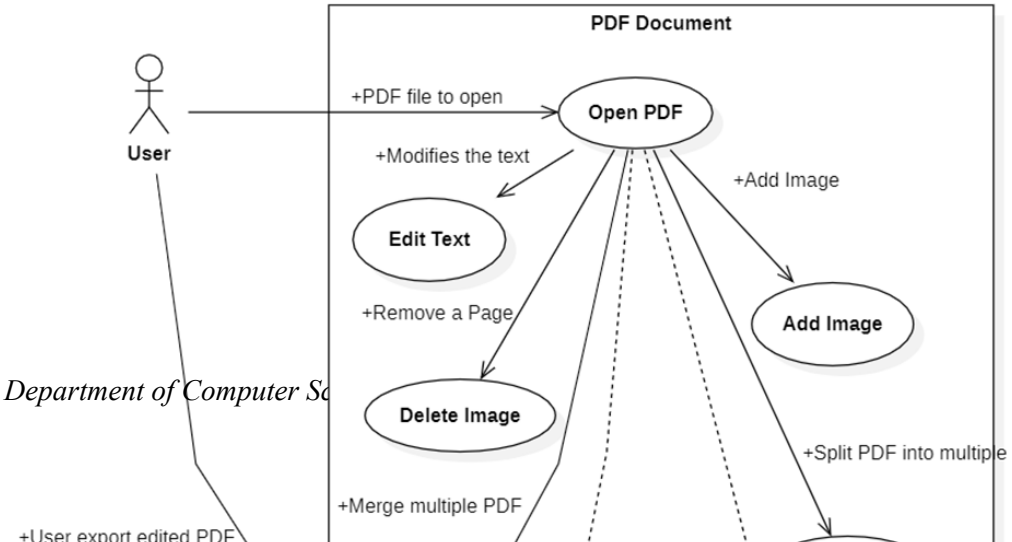
The following list style is the sample to consistently follow in the whole report.

- List items 1
- List items 2

3. Requirement Analysis

The following parts of Software Requirements Specification (SRS) report should be included in this chapter

Use Cases Diagram(s)



3.1. Detailed Use Case

Open PDF

The user selects a PDF file to open in the editor.

Edit Text

The user modifies the text within the PDF document.

Add Image

The user adds an image to the PDF document.

Delete Page

The user removes a page from the PDF document.

Split PDF

The user splits the PDF document into multiple documents.

Merge PDF

The user merges multiple PDF documents into a single document.

Annotate

The user adds annotations, such as notes or highlights, to the PDF document.

Export PDF

The user exports the edited PDF document in a desired format, such as .pdf or .docx.

3.2. Functional Requirements

- View and edit PDF files: The application should be able to open, view, and edit PDF documents. This may include the ability to add, delete, or modify text and images, as well as to rearrange or delete pages within the document.
- Annotate PDF files: The application should allow users to add annotations to PDF documents, such as highlighting, underlining, or adding notes or comments.
- Convert PDF files: The application should be able to convert PDF documents to other file formats, such as Word or Excel, and vice versa.
- Merge and split PDF files: The application should allow users to merge multiple PDF documents into a single file, or split a single PDF document into multiple files.
- Protect PDF files: The application should provide features to password-protect PDF documents, as well as to restrict editing or printing.

- Customize PDF files: The application should allow users to customize the appearance of PDF documents, such as by adding watermarks or adjusting the layout.
- Support for various PDF versions: The application should support different versions of the PDF specification, such as PDF 1.7 and PDF 2.0.
- User-friendly interface: The application should have a user-friendly interface that is easy to navigate and use. It should also have features such as search and find, as well as the ability to bookmark pages and add notes.

3.3.Non-Functional Requirements

- Usability: The application should be easy to use and understand, with a clear and intuitive interface.
- Performance: The application should open, save, and manipulate PDF files efficiently, with minimal delays or errors.
- Compatibility: The application should support a wide range of PDF versions and file formats, and be able to import and export documents with minimal loss of data.
- Security: The application should protect the privacy and security of user data, and prevent unauthorized access to or manipulation of PDF documents.
- Reliability: The application should be stable and reliable, with minimal crashes or errors.
- Scalability: The application should be able to handle large PDF files and a large number of users without degrading performance.
- Maintainability: The application should be easy to maintain and update, with a clear and modular design.
- Internationalization: The application should support multiple languages and localization options.
- Accessibility: The application should be accessible to users with disabilities, and comply with relevant accessibility standards.

4. Design and Architecture

The following parts of Software Design Description (SDD) report should be included in this chapter.

System Architecture

The system architecture of SmartEditor, a mobile application for PDF manipulation and editing, involves multiple components that work together to provide a seamless user experience. The application is designed to run on both Android and iOS platforms and is built using a combination of structural and object-oriented programming methodologies.

The architecture of the SmartEditor application can be divided into three main components: the user interface, the backend server, and the database. The user interface is responsible for displaying the application's features and functionalities to the user. The backend server handles the processing of user requests and performs the necessary operations on the PDF files. The database stores user information and PDF files for easy access and retrieval.

The user interface of SmartEditor is designed to be intuitive and user-friendly, with a modern and sleek design. Users can navigate through the application's features using an easy-to-use menu system. The main screen of the application displays a list of PDF files available for editing, along with the necessary tools and options for editing and manipulating the files.

The backend server of SmartEditor is responsible for performing the necessary operations on PDF files, such as merging, splitting, and adding or removing pages. The server uses various algorithms and techniques to ensure that the operations are performed quickly and efficiently.

4.1. Data Representation [Diagram + Description]

Document data: This includes the text content of the document, as well as any metadata associated with the document, such as the date of creation and last modification.

Formatting data: This includes information about the formatting of the document, such as font style, size, and color.

Revision data: This includes information about previous versions of the document, such as the date of each revision and the changes made.

4.2. Process Flow/Representation

4.3. Design Models [along with descriptions]

Object-Oriented Design Model: This design model is based on the concept of objects, which encapsulate data and functionality. This model focuses on creating reusable, modular, and scalable code by breaking down the application's functionality into smaller objects that communicate with each other. This design model can help in the development of the Smart Editor app by encapsulating the editing functionality into objects that can be reused and extended.

Service-Oriented Architecture (SOA) Design Model: This design model is an architectural approach that decomposes an application into a set of services that can be loosely coupled, independently deployed, and easily consumed by other applications. This design model can help in the development of the Smart Editor app by creating editing services that can be accessed by other applications, making it easy to integrate the Smart Editor app into a larger software ecosystem.