

# **BillForge KUET Teachers' Edition**

## **(An Automatic Bill Generator from Doc file)**

By

**Chinmoy Modak Turjo**

Roll: 1907003

&

**Swaraj Chandra Biswas**

Roll: 1907069



**Supervisor:**

**Dr. Sheikh Imran Hossain**

Assistant Professor

Dept. of Computer Science and Engineering

Khulna University of Engineering & Technology

---

Signature

**Department of Computer Science and Engineering**

**Khulna University of Engineering & Technology**

**Khulna 9203, Bangladesh**

**November 2023**

## **Acknowledgements**

We, Chinmoy Modak Turjo and Swaraj Chandra Biswas, submit our project report, "BillForge KUET Teachers' Edition (An Automatic Bill Generator from Doc File)", which is a crucial part of our CSE 3200: System Project Development Project coursework.

We would first want to express our sincere gratitude to Dr. Sheikh Imran Hossain, an assistant professor in the computer science and engineering department. His vast knowledge and direction in this area have been essential to the success of our endeavor. We sincerely appreciate his intellectual guidance, unwavering support, and energetic oversight. His astute suggestions and helpful critiques have greatly influenced the direction of this endeavor. The completion of this effort would not have been possible without the invaluable encouragement and enthusiasm of Dr. Sheikh Imran Hossain.

Finally, but most definitely not least, we give thanks to God for bestowing upon us the skills and aptitudes that allowed us to start this project and see it through to completion.

**Authors**

## **Abstract**

A desktop application called "BillForge KUET Teachers' Edition" was created especially for the office executives of the several departments at Khulna University of Engineering & Technology (KUET). Its main goal is to make the process of creating bills for each teacher at the conclusion of each semester or term as efficient as possible. Users, the main controllers in this system, log in with their specific user ID and password. After logging in, users can choose the necessary doc file, which is the primary data source for the system and contains charts or tables with teacher information. Teachers engaged in academic activities for a certain year or semester are listed in a doc file that is typically prepared by the department office at KUET at the end of each semester. The main difficulty is creating separate bills for every teacher using this document; this is a process that takes a lot of time. Executives in charge of offices have had to devote a lot of time on gathering and mapping instructor data to the appropriate Excel files. The bill creator also has to manually enter data into the Excel sheets before sending emails to every teacher, which is a time-consuming and error-prone operation. By automating the billing system and the emailing process that follows, this project seeks to eliminate these inefficiencies and drastically reduce manual labor and error-proneness.

# Table of Contents

	<b>Page</b>
Acknowledgements	ii
Abstract	iii
List of Figures	vi
<b>1 Introduction</b>	<b>1</b>
1.1 Background	1
1.2 Objectives	1
1.3 Scope	2
1.4 Unfamiliarity of the problem	2
1.5 Project Planning	3
<b>2 Related Works</b>	<b>3</b>
2.1 Related Works	3
2.2 Limitation of Existing Solutions	4
<b>3 System Design</b>	<b>5</b>
3.1 System Overview	5
3.2 Requirement Analysis	6
3.3 System Analysis	7
3.4 System Flow Diagram	9
3.5 Use Case Diagram	10
3.6 Data Flow Diagram	11
3.7 Description of Necessary Tools for this system	11
3.7.1 Custom build generalized Flask API for public use:	12
3.7.2 SQLite Database:	13
3.7.3 Python-docx Library	14
3.7.4 Figma	14
3.7.5 Tkinter Library	15
3.7.6 Figma API	16
3.7.7 OpenPyXL Library	17
3.7.8 Google Translator	18
3.7.8 SMTP lib Module	19
<b>4 Project Implementation</b>	<b>20</b>
4.1 System Implementation	20

4.2 Deployment	28
4.3 Ethical & Moral Issues	29
4.4 Financial Analyses and Budget	29
<b>5 Conclusion</b>	<b>30</b>
5.1 Conclusion & Challenges Faced	30
5.2 Future Study	31
<b>6 References</b>	<b>31</b>

## List of Figures

	Page
1.Gantt Chart	3
2.Spiral Model	6
3.System Flow Diagram	9
4.Use Case Diagram	10
5.Data Flow Diagram	11
6.Flask API	12
7.SQLite Database	13
8.Python-docx	14
9.Figma	14
10.Tkinter	15
11. Figma API	16
12. OpenPyXL	17
13.Google Translator	18
14.SMTP Lib Module	19
15.Welcome GUI	20
16.Signup Screen	21
17.Bill Generator	21
18.All Details of Bill Generator	22
19.Bill Generation in progress	22
20.New Bill Doc	23
21.Template Excel	24
22.After Generating Bill	25
23.Empty Email Check Sheet	26
24.After Filling Email Address	26
25.Email Sending in progress	27
26.Email Received	27
27.Bill Summary	28

# **1 Introduction**

## **1.1 Background**

The application is a desktop-based, cross-platform tool capable of running on both UNIX and Windows operating systems. To achieve this cross-platform functionality, we have utilized Python, a robust and object-oriented programming language. Python is known for its simplicity, readability, dynamic integration capabilities, being open-source, and free to use. It supports a wide range of frameworks and libraries, which are crucial for this project's implementation.

The design of this system leverages the principles of object-oriented programming (OOP). We have utilized various Python libraries to fulfill the project's requirements, including 'python-docx' for reading data from DOC files, 'openpyxl' for reading and writing Excel files, 'smtplib' for email functionalities, and 'Tkinter' for Graphical User Interface (GUI) related tasks. Additionally, the Google Translator API is employed for translating text from English to Bangla.

To successfully implement this project, a strong understanding of object-oriented programming concepts is essential. Knowledge of API usage and integration of various libraries and modules is also crucial to ensure seamless design and functionality of the project.

## **1.2 Objectives**

- To facilitates the work of office executives of different departments of KUET
- To save work hours of different offices of KUET
- To automate the process of creating bills and sending emails
- To reduce the chance of human errors
- To increase the efficiency of executives thus they can do more work in the limited time scope

- To take a step forward towards automating KUET
- To develop a system that can automate the work culture of departments to some extent

### **1.3 Scope**

The 'BillForge' application is a desktop tool designed to automate the process of bill creation for the teachers at Khulna University of Engineering & Technology (KUET) and facilitate the sending of these bills via email. At the end of each semester, the administrative offices of various departments at KUET are tasked with generating bills for teachers who have participated in academic activities related to specific batches of students. These bills, formatted in Excel, incorporate pre-designed rules for calculating dues based on various factors, such as the number of answer scripts evaluated by a teacher for a particular year or the number of credits taught in sessional courses etc.

The primary users of this application are the administrative offices of KUET's respective departments. With the aid of 'BillForge,' these offices can efficiently generate individualized bills for each teacher in Excel format and then directly email these bills to the respective teachers.

### **1.4 Unfamiliarity of the problem**

As of till now, there is no automation solution to the creation of billing process of teachers. Rather there are some tweaks to help the biller to create bills. This system properly addresses this issue of automation and fulfills the need of automation of the billing process. The idea is drawn from observing the workflow the different offices in Khulna University of Engineering & Technology.



## 1.5 Project Planning

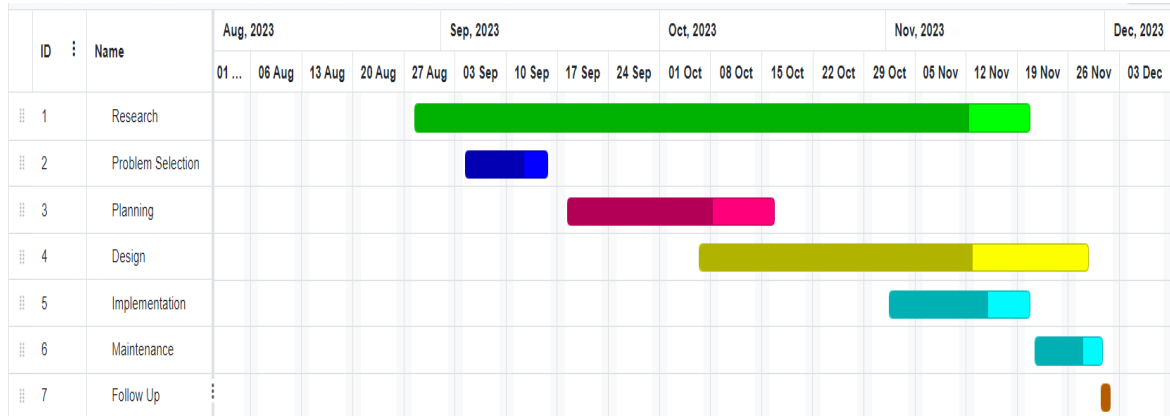


Fig-1.5 Gantt Chart of System Development Project

## 2 Related Works

### 2.1 Related Works

#### University of California, Irvine (UCI):

UCI has implemented a system called "**eCampus**" that automates a variety of tasks, including student registration, grading, and financial aid processing. The system has saved the university an estimated \$10 million per year in labor costs.

#### University of Cambridge:

The University of Cambridge has developed a system called "**CamSIS**" that provides a single platform for managing all aspects of student administration. The system has been credited with improving the efficiency and accuracy of student records

#### National University of Singapore (NUS):

NUS has implemented a system called "**ERPNext**" that automates a variety of tasks, including procurement, inventory management, and human resources. The system has been credited with improving the efficiency of NUS's administrative operations.

#### **Enterprise Resource Planning (ERP) Systems:**

ERP systems like SAP, Oracle, and Microsoft Dynamics are used by many organizations to integrate and automate various business processes, including finance, human resources, and procurement

#### **Document Management Systems:**

Systems like SharePoint, Documentum, and Alfresco help organizations manage, organize, and automate document-related processes.

#### **Email Automation Tools:**

Tools such as Microsoft Power Automate, Zapier, and Integromat enable automation of email workflows, including sending emails triggered by specific events or conditions.

#### **Workflow Automation Platforms:**

Workflow automation tools like Nintex, and Appian are designed to automate and streamline business processes, reducing manual intervention and improving overall efficiency.

## **2.2 Limitation of Existing Solutions**

The main problem of existing solution is that they are mostly university or institution centric rather than a general solution. Billing systems of different institutions vary from each other. This prevents programmers or inventors to develop a general open for all system solution. To work on a particular and scoped field gives more room for customization and edit ability.

### **3 System Design**

The primary objective of this project is to enhance the efficiency of administrative tasks at Khulna University of Engineering & Technology (KUET). By implementing an automated system for bill generation and payment processing, we aim to facilitate streamlined operations and effective communication within the university's administrative framework. This system is equipped with features such as automated data validation to minimize errors, along with secure data storage and access control for authorized personnel only.

Such advancements are expected to result in substantial time and resource savings across KUET's various departments, boosting staff productivity and efficiency. This aligns with KUET's strategic goal of transitioning into a digitally-advanced and modern educational institution.

Tailored to meet the specific needs of executives in different departments, with an initial implementation in the Computer Science and Engineering (CSE) department, the system is designed to enhance clarity and transparency in bill processing. The deployment of this project promises an environment with reduced errors, thereby supporting KUET staff in achieving a more efficient and effective administrative workflow.

#### **3.1 System Overview**

The “BillForge KUET Teachers’ Edition” system is developed using the spiral software development life cycle model. It is also referred to as a risk handling model. In the diagrammatic representation it looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a Phase of software development process. The exact number of phases needed to develop the product can varied by the project manager depending upon the risk. As the project manager has an important role to develop a product using the spiral model.

The spiral model of a Software Development Life Cycle (SDLC) model that provides a systematic and iterative approach to software development. It is based on the idea of a

spiral, with each iteration of the spiral representing a complete software development cycle, from requirement gathering and analysis to design, implementation, testing & maintenance.

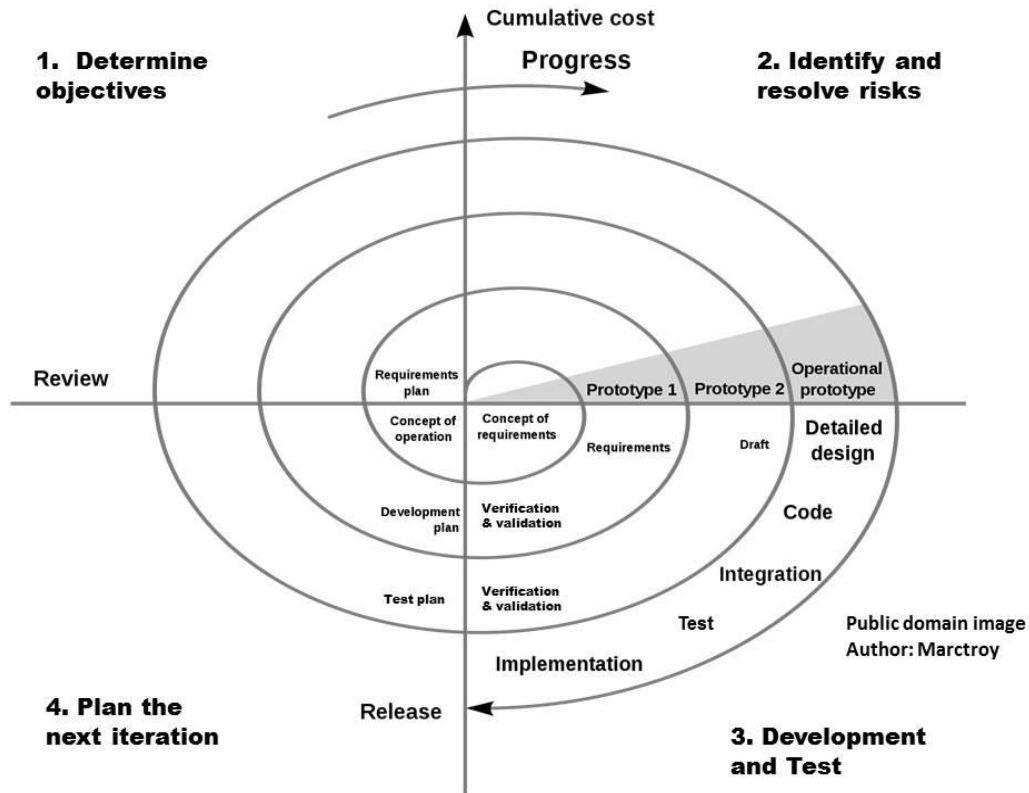


Fig-3.1: Spiral Model SDLC

### 3.2 Requirement Analysis

The Requirement Analysis for the 'BillForge KUET Teachers' Edition' project entails a thorough examination of the specific needs of office executives across various departments at Khulna University of Engineering & Technology (KUET). This process begins with gaining a comprehensive understanding of the bill creation procedures, which vary across different departments. The analysis also includes a detailed review of email communication processes and the existing workflow in each department.

It is crucial to identify and clearly document the essential functionalities needed for automation, ensuring the system operates seamlessly and efficiently. The analysis should establish measurable criteria for success, illustrating how the system will contribute to

saving work hours, enhancing efficiency, and minimizing the likelihood of errors. Factors such as compatibility with current systems and technologies, as well as the implementation of robust security measures to safeguard sensitive data during bill creation and transmission, are vital considerations.

Furthermore, the Requirement Analysis must consider the system's adaptability to the 'Smart Digital KUET' vision, ensuring it aligns with the broader objective of improving the work culture and efficiency across departments.

### 3.3 System Analysis

There are two types of people who will be using this software.

1. The User or office staff
2. The programmer who will maintain it

**User:** The office staff will be the prime user of this program who will use the system to generate the billing for the teacher, selecting the docs, selecting the excel files and sending emails.

**Programmer:** The one who designed the system and in time of need maintains them and do modification and upgradation work.

As illustrated in Figure 3.4, the user initiates the system by double-tapping the application icon. This action leads to the system's start flow. The user is then directed to a login screen, where they can access the system using a unique user ID and password. Successful authentication recognizes the user as an authenticated member, granting access to the control screen. To calculate the value of every teacher's bill there is a custom developed API (Application Programming Interface) which is responsible for translating the value of taka amount in Bengali corresponding words. The user info and password will be saved in the database. Besides the total summary of a semester's expense for teachers' bill is calculated separately into an excel file and saved on the database.

On the control screen, users have the ability to select the source file, which contains the exam bill information in a document format. Additionally, users can specify the destination

location for the output files. Another critical feature allows the user to choose the email file, which contains the mailing addresses of the teachers. Following the generation of individualized Excel sheets for each teacher, the user has the capability to send emails to the respective teachers directly from the system. This process streamlines the task of distributing the generated bills, ensuring efficiency and accuracy.

The illustration in Figure 3.5, is the use case diagram. Where the actor is the user. The user's first action is login. After login, the user will navigate to the bill generator screen, there the user will get 3 options:

1. Select the Destination Folder where the data will be stored.
2. The email picker xlsx file which will contain the email list of the teachers
3. The doc picker which will contain the doc of bill of the teachers of a semester or term.

In order to complete the task, the user has to select the 3 options. The main goal is to generate the bill, the user has to perform option 2 & 3. After the generation of bill, the user can check the email file and write the email addresses of the respective teachers. Then at the end, the user can send mail or right after the generation of bills the user can terminate the window and close the program. This send and check email feature is optional here.

The illustrated figure-3.6 is data flow diagram, This depicts how the data is passed between different modules and features of the system. The user can signup on the system and login to the system. The user id and password are saved on the database. Any wrong input can generate error message. After successful login, the system user selects the path of the necessary files and pick files through windows default file picker. After the clicking of generate bill button, the program creates bill for every individual teacher. When the user press email send button a mail request is sent to the email sending protocol which will be responsible for email sending request handling. Then the user can repeat the process if he/she wants.

### 3.4 System Flow Diagram

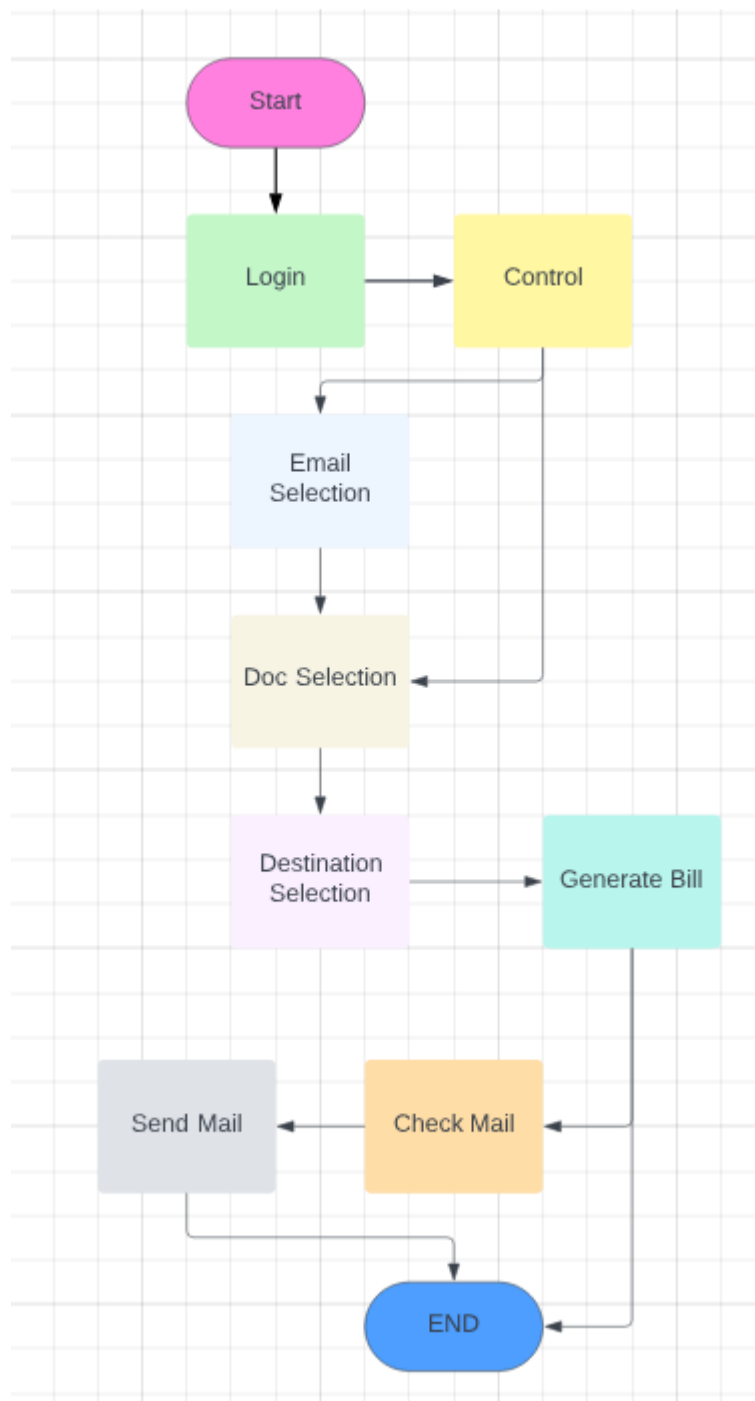


Fig-3.4: System Flow Diagram of BillForge

### 3.5 Use Case Diagram

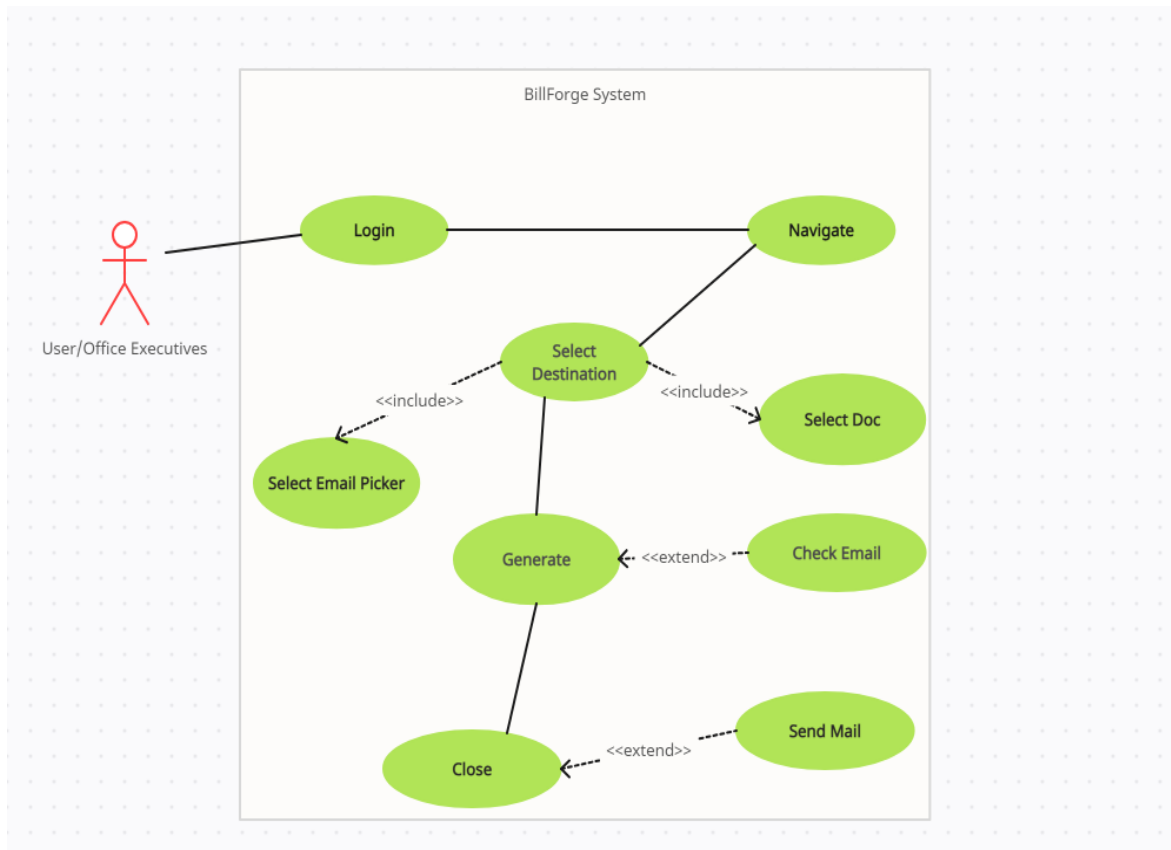


Fig-3.5: Use Case Diagram of BillForge



### 3.6 Data Flow Diagram

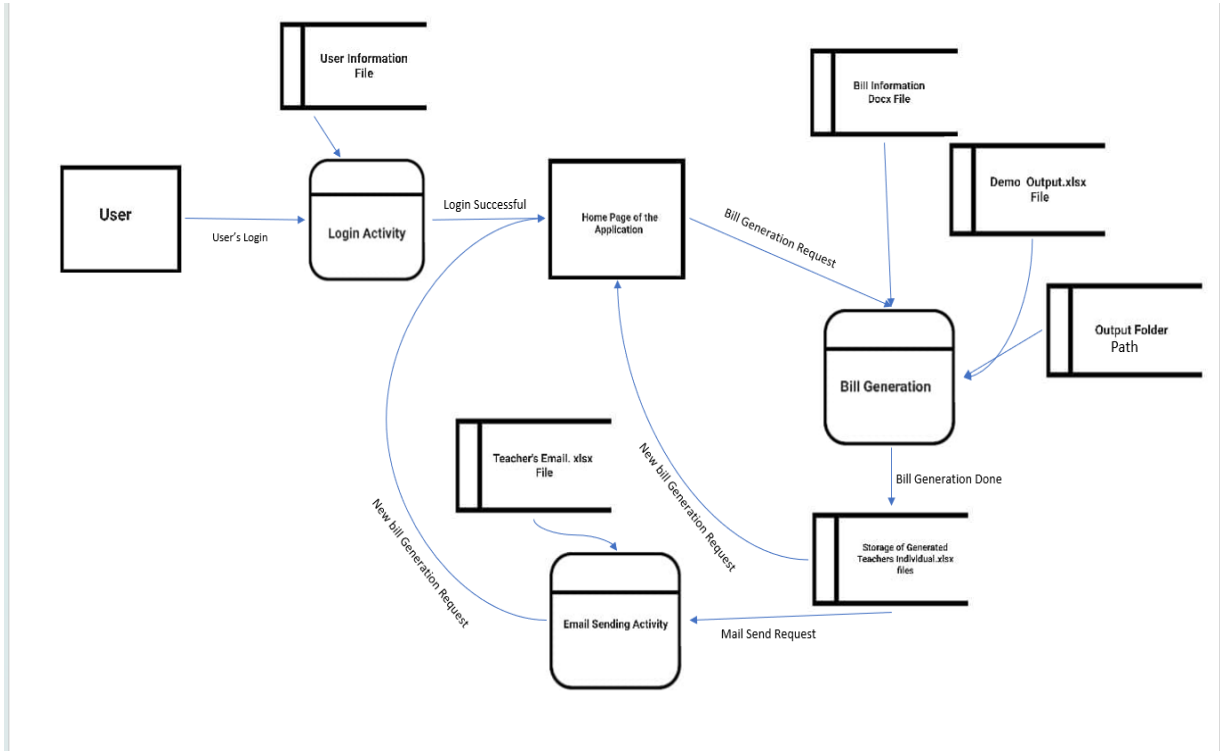


Fig-3.6: Data Flow Diagram of BillForge

### 3.7 Description of Necessary Tools for this system

To develop this application, we need an authentication system and database to store user id and password such that any unauthorized user cannot get access into the system and does any unwanted job to the system. To give the user a good experience we added a GUI application on top of our system. GUI can generally give more good navigation & usability. This system works on a document file with a common billing layout for the teachers of the KUET and thus produces excel file sheet for any individual teacher generating their bill summary for a corresponding term or semester. As this system reads information from docx file, we need some sort of libraries which can work & extract values from doc files and as we are putting the values back into an excel sheet we need some libraries to work on excel file as well. To convert the Bangla taka to corresponding words we have created a Flask API and it is hosted

online and as it is a public API so anyone can use it for their purpose. For many different features implementation of this system, we need some more libraries and Api calls.

### 3.7.1 Custom build generalized Flask API for public use:



Fig-3.7.2 Flask API

**Flask** is a lightweight web application framework for Python. It is designed to be simple, easy to use, and flexible. Flask is part of the microframework category, which means it doesn't come with a lot of built-in features like some larger frameworks such as Django. Instead, Flask provides the essentials for building web applications and allows developers to choose and integrate the components they need. The custom build API will take input a string corresponding the Bangla Taka to Bangla written words.

Features of Flask Framework:

- Routing
- HTTP Request Handling
- Extensions
- Development Server
- Microframework etc.

### 3.7.2 SQLite Database:



Fig-3.7.2: SQLite Database

**SQLite** is an embedded, server-less relational database management system. It is an in-memory open-source library with zero configuration and does not require any installation. Also, it is very convenient as it's less than in size, which is significantly lesser than other database management systems. In this database, the user id, password and summary result of a term is stored.

Features of SQLite Database:

- Serverless & Self-contained
- Zero Configuration
- Cross-Platform
- ACID Compliance
- Community Support

### 3.7.3 Python-docx Library

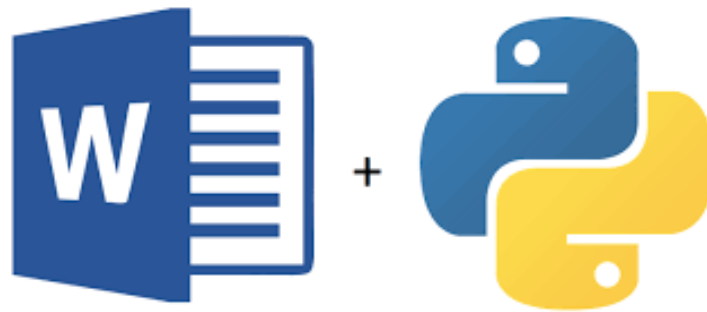


Fig-3.7: Python Docx

**Python-docx** is a Python library for creating and updating Microsoft Word (.docx) files. It allows you to work with Word documents, including creating new documents, modifying existing ones, and extracting information from them. The library provides a simple and convenient way to generate Word documents programmatically.

Key Features of python-docx library:

- Document Creation
- Document Modification
- Text Formatting
- Paragraph & Run Styles
- Tables
- Headers & Footers

### 3.7.4 Figma



Fig-3.7.4: Figma

**Figma** is a cloud-based design and prototyping tool used in user interface (UI) and user experience (UX) design. It enables designers to collaborate in real-time on the creation of digital designs for websites, mobile applications, and other interactive interfaces. Figma is popular for its collaborative features, accessibility from any device with an internet connection, and the ability to streamline the design-to-development workflow.

Features of Figma:

- Cloud Based Collaboration
- Vector Editing
- Component Based Design
- Prototyping and Interaction Design etc.

### 3.7.5 Tkinter Library



Fig-3.7.5 Tkinter Library

**Tkinter** is the standard GUI (Graphical User Interface) toolkit that comes with Python. It is used to create desktop applications with graphical interfaces. Tkinter is a thin object-oriented layer on top of the Tcl/Tk GUI toolkit. It's simple to use and allows developers to create windows, dialogs, buttons, textboxes, and other GUI elements.

Key features of Tkinter library are:

- Simple & Lightweight
- Integration with Python
- Even-Driven Programming
- Cross-Platform

### 3.7.6 Figma API

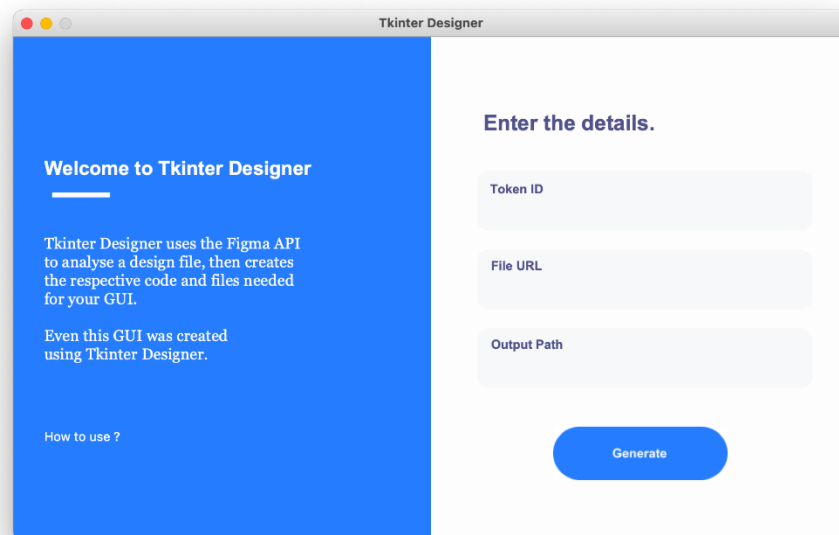


Fig-3.7.6: Tkinter Designer

We have used **Tkinter Designer** as Figma API caller to translate Figma design to corresponding tkinter python code. It will receive the Figma token ID, Figma file URL and destination path of where the data will be saved.

### 3.7.7 OpenPyXL Library



Fig-3.7.7: OpenPyXL Library

**Openpyxl** is a Python library for reading and writing Excel (xlsx) files. It allows you to work with Excel files (both reading from and writing to) using Python. This library is particularly useful for tasks such as data analysis, data manipulation, and automating Excel-related processes.

Key features of openPyXL libraries are:

- Reading & Writing Excel Files
- Support For Excel Worksheets & Cells
- Cell Formatting
- Charts & Images
- Data Validation
- Formula Support

### 3.7.8 Google Translator



Fig-3.7.8 Google Translator

Google Translator, or Google Translate, is a free online language translation service developed by Google. It allows users to translate text, documents, web pages, and even spoken words or phrases between different languages. Google Translate supports a wide range of languages, making it a versatile tool for communication and information exchange across language barriers.

Key features of Google Translator are:

- Text Translation
- Document Translation
- Auto Detection of Language
- Synonyms & Definitions



### 3.7.8 SMTP lib Module

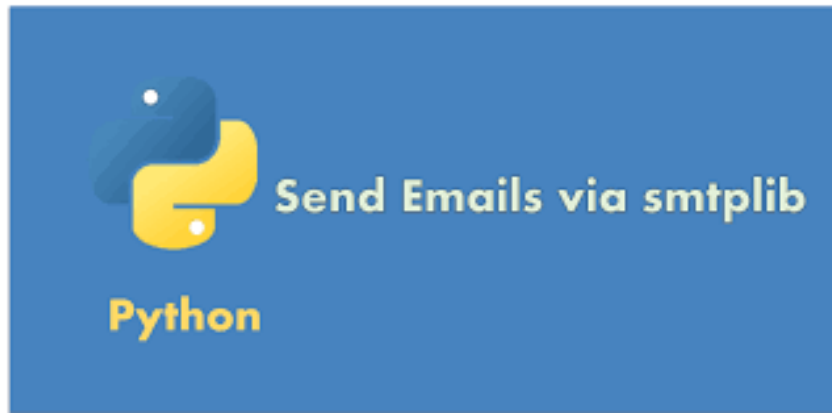


Fig-3.7.8 SMTP lib Module

**Smtplib** is a module in Python that defines an SMTP (Simple Mail Transfer Protocol) client session object used to send email to any internet machine with an SMTP or ESMTP listener daemon process running. SMTP is a protocol used for sending emails between servers, and smtplib provides a way for Python scripts to communicate with an SMTP server for sending email messages.

Key Features of STMP lib module:

- Sending Emails
- Support for SSL/TLS
- Authentication
- Integration with Email Message

## 4 Project Implementation

### 4.1 System Implementation

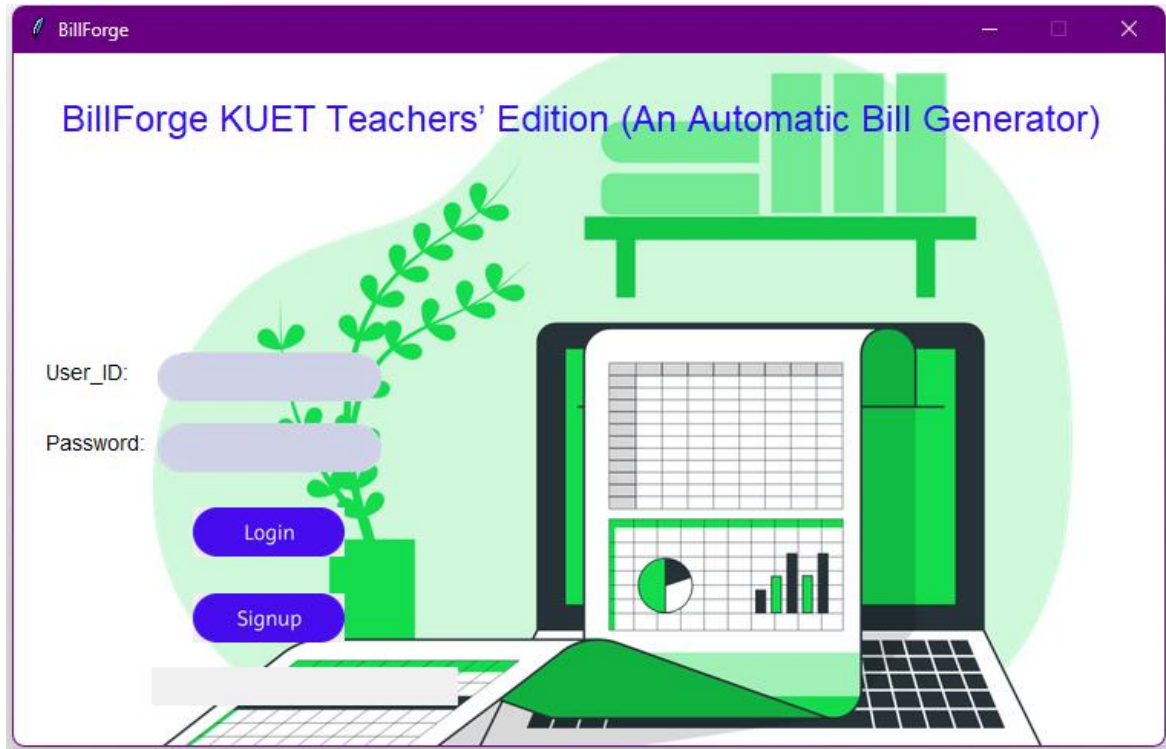


Fig-4.1: Welcome GUI

The image shows a web browser window titled "BillForge". The page has a light green background. At the top center is the Khulna University of Engineering & Technology logo, which is a green shield with a white book and a lamp inside, and Bengali text around it. Below the logo, the text "Khulna University of Engineering & Technology" is displayed. Underneath, there are two input fields: "Username:" and "Password :". Below these fields is a button labeled "SignUp".

Fig-4.2 Signup UI

The image shows a web browser window titled "BillForge". The page has a light green background. At the top center is an illustration of a man in a green shirt sitting at a desk, looking at a computer monitor. The monitor displays a document labeled "INVOICE". Below the illustration, the text "Bill Generator" is displayed. Underneath, there are three input fields: "Select Bill Docx File:", "Select Excel File:", and "Select Destination :". To the right of each input field is a button: "Open .docx File", "Open Email File", and "Open Folder". At the bottom, there are three buttons: "Check Email", "Generate Bill", and "Send Email".

Fig-4.3: Bill Generator Screen

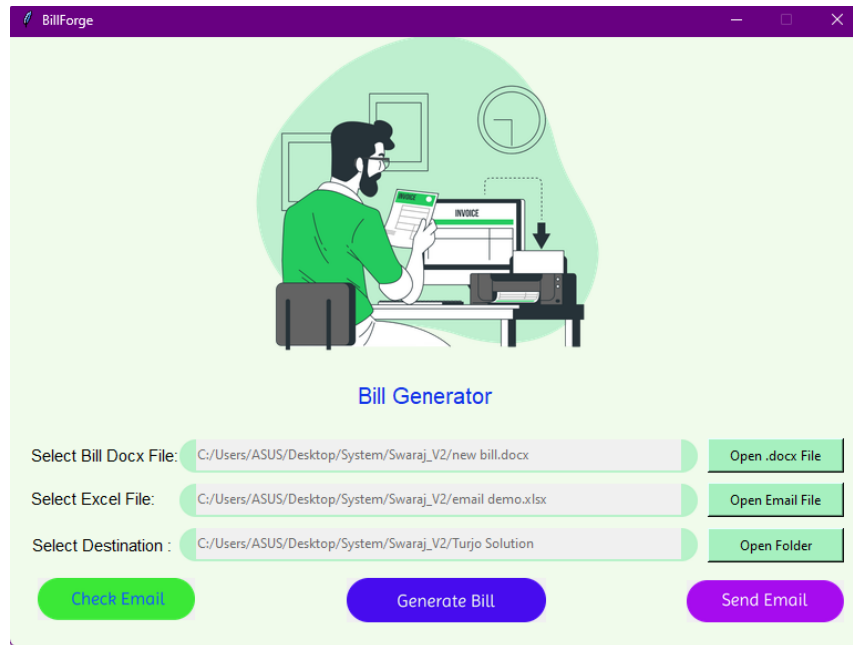


Fig-4.3: Bill Generator Screen after filling all details

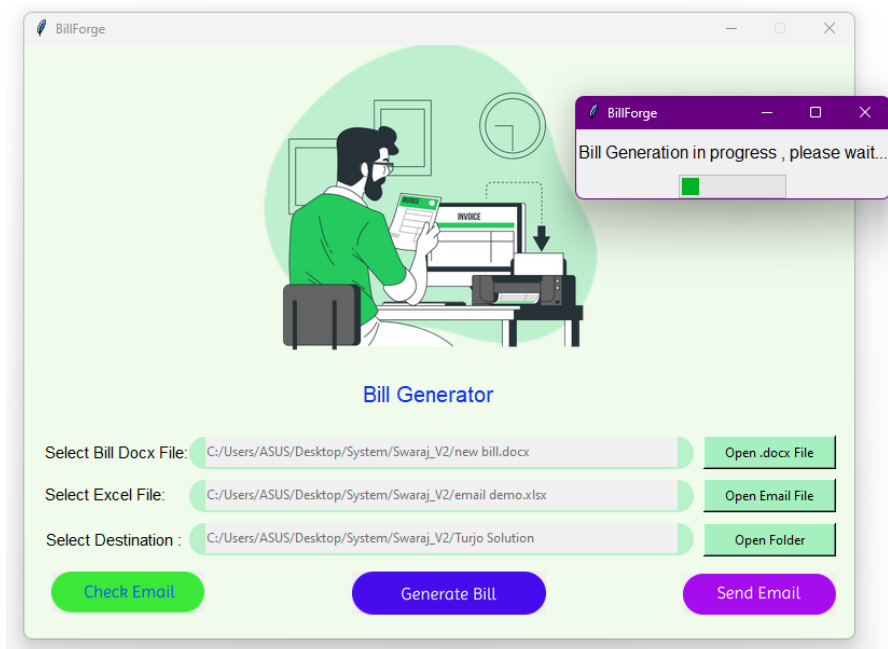


Fig-4.4: Bill Generator in progress



1

2

3

4

5

6

খুলনা প্রকৌশল ও প্রযুক্তি বিশ্ববিদ্যালয়

পরীক্ষা সংক্রান্ত কাজের বিল ফর্ম

নিয়মিত পরীক্ষা

বর্ষ:টার্ম:

বিভাগ:

পরীক্ষা সংক্রান্ত কাজের বিবরণ

ক্র. নং	কাজের নাম	পরীক্ষার নাম	বছর	বিষয়	পত্র	যাত্রা পত্রের সংখ্যা	অর্থপূর্ণ	টাকা	Rates
০১.	প্রশ্নপত্র প্রণয়ন							০.০০	Per Question : 3600
০২.	সভাপতি, পরীক্ষা কমিটি							০.০০	Per Term : 3000
	প্রশ্নপত্র নিয়ামন							০.০০	Max : 4500 Min : 1500
	উত্তরপত্র পরীক্ষণ							০.০০	Per Section : 60, (Min : 750)
০৩.	অনলাইন ভাইভা							০.০০	Per Student Per Teacher: 50
	ক্লাস টেস্ট							০.০০	Per Class Test Per Student : 45
	সফটওয়্যার প্রজেক্ট							০.০০	Per Project Per Stud : 600
	সেমিনার							০.০০	Per Stud : 300
০৪.	ব্যবহারিক							০.০০	(1.5crh)Per Stud Per Teacher : 300
	সেন্ট্রাল ভাইভা							০.০০	Per Student : 150
	থিসিস/প্রজেক্ট পরীক্ষণ							০.০০	Per Project Per Examiner : 2400
	থিসিস/প্রজেক্ট প্রোগ্রেস							০.০০	Per Student Per Examiner : 100
০৫.	নিরূপণ							০.০০	Per Student Per Examiner : 225
	থিসিস/প্রজেক্ট ফাইনাল							০.০০	Per Thesis/ Project : 4500
	নিরূপণ							০.০০	Per Stu Per Tab : 45 (Min 750)
০৬.	টেবুলেশন							০.০০	Per Student : 20
	টেবুলেশন ভেরিফিকেশন							০.০০	Per Section Script : 10 (Min 1000)
০৭.	স্ক্রিপ্টনি	বি. এসসি. ইঞ্জি.	২০২২					০.০০	Per Hour : 900
০৮.	ইনভিজিলেশন	বি. এসসি. ইঞ্জি.	২০২২					০.০০	Per Question : 1500
০৯.	প্রশ্নপত্র প্রস্তুতকরণ	বি. এসসি. ইঞ্জি.	২০২২					০.০০	Per Student : 900
	ভেরিফিকেশন অব ফাইনাল প্রজেক্টেশন							০.০০	Per Student/Per Term : 225
	গ্রাজুয়েট প্রজেক্ট							০.০০	
১১.	প্রজেক্ট/টার্ম পেপার							০.০০	
১২.	কোর্স সমন্বয়কারী (প্রতি টার্ম)							০.০০	Per Term : 3600
			রাজস্ব ডিস্কট			সর্বমোট		০.০০	
প্রতিষ্ঠানকর,									
সভাপতি, পরীক্ষা কমিটি									স্বাক্ষর
									সীল
বি.স্র:- প্রত্যেক পরীক্ষার জন্য পৃথক বিল করতে হবে।									
অফিসে ব্যবহারের জন্য									
পরীক্ষক									
পরীক্ষা নিয়ন্ত্রক									
প্রস্তুতকারী: পরীক্ষাকারী: সুপারিশকারী: অনুমোদনকারী:									

Fig-4.6: Copy Excel File Before Running BillForge

খুলনা প্রকৌশল ও প্রযুক্তি বিশ্ববিদ্যালয়									
পরীক্ষা সংক্রান্ত কাজের বিল ফর্ম									
নাম: ড. শেখ ইমরান হোসেন			নিয়মিত পরীক্ষা ২০২২						
পদবী: সহকারী অধ্যাপক			বর্ষ: ৪র্থ		টার্ম: ১ম				
বিভাগ/শাখা: সি.এস.ই			বিভাগ: সি.এস.ই						
পরীক্ষা সংক্রান্ত কাজের বিবরণ									
ক্র : নং	কাজের নাম	পরীক্ষার নাম	বছর	বিষয়	পত্র	খাতা #প ত্রের সং খ্যা	অর্থ / পূর্ণ	টাকা	Rates
০১.	প্রশ্নপত্র প্রণয়ন	বি. এসসি. খঞ্জি:	২০২২					০.০০	Per Question : 3600
০২.	সভাপতি, পরীক্ষা কমিটি	বি. এসসি. খঞ্জি:	"					০.০০	Per Term : 3000
	প্রশ্নপত্র নিয়ামন	বি. এসসি. খঞ্জি:	"					০.০০	Max : 4500 Min : 1500
	উত্তরপত্র পরীক্ষণ	বি. এসসি. খঞ্জি:	"			০		০.০০	Per Section : 60, (Min : 750)
০৩.	অনলাইন ভাইজা	বি. এসসি. খঞ্জি:	"					০.০০	Per Student Per Teacher: 50
	ক্লাস টেস্ট	বি. এসসি. খঞ্জি:	"			০	১	০.০০	Per Class Test Per Student : 45
	সফটওয়্যার প্রজেক্ট	বি. এসসি. খঞ্জি:	"					০.০০	Per Project Per Stud : 600
০৪.	সেমিনার	বি. এসসি. খঞ্জি:	"			২৭	২৫	৩২৪.০০	Per Stud : 300
	ব্যবহারিক	বি. এসসি. খঞ্জি:	"			০		০.০০	(1.5scrh) Per Stud Per Teacher : 300
	সেন্ট্রাল ভাইজা	বি. এসসি. খঞ্জি:	"			০	১১	০.০০	Per Student : 150
	খিসস/প্রজেক্ট পরীক্ষণ	বি. এসসি. খঞ্জি:	"					০.০০	Per Project Per Examiner : 2400
	খিসস/প্রজেক্ট প্রোগ্রাম	বি. এসসি. খঞ্জি:	"			৪০	25	৪০০০.০০	Per Student Per Examiner : 100
০৫.	খিসস/প্রজেক্ট ফাইনাল	বি. এসসি. খঞ্জি:	"					০.০০	Per Student Per Examiner : 225
	খিসস/প্রজেক্ট	বি. এসসি. খঞ্জি:	"					০.০০	Per Thesis/ Project : 4500
০৬.	টেবুলেশন	বি. এসসি. খঞ্জি:	"					০.০০	Per Stu Per Tab : 45 (Min 750)
	টেবুলেশন ভেরিফিকেশন	বি. এসসি. খঞ্জি:	"			০		০.০০	Per Student : 20
০৭.	স্ক্রুটিনি	বি. এসসি. খঞ্জি:	"			০		০.০০	Per Section Script : 10 (Min 1000)
০৮.	ইনভিজিলেশন	বি. এসসি. খঞ্জি:	"			২		৫৪০০.০০	Per Hour : 900
০৯.	প্রশ্নপত্র প্রস্তুতকরণ	বি. এসসি. খঞ্জি:	"			২		৩০০০.০০	Per Question : 1500
	ভেরিফিকেশন অব	বি. এসসি. খঞ্জি:	"						
১০.	ফাইনাল প্রাজেকশন	বি. এসসি. খঞ্জি:	"			১		৯০০.০০	Per Student : 900
	গ্রাউন্ডভাইজার	বি. এসসি. খঞ্জি:	"			০		০.০০	Per Student/Per Term : 225
১১.	প্রজেক্ট/টার্ম পেপার	বি. এসসি. খঞ্জি:	"						
১২.	কোর্স সমন্বয়কারী (প্রতি টার্ম)							০.০০	Per Term : 3600
কম্বা: তেরো হাজার ছয়শত চব্বিশ টাকা মাত্র						সর্বমোট		১৩৬২৪.০০	
প্রতিষ্ঠান,									
সভাপতি, পরীক্ষা কমিটি						স্বাক্ষর	সীল		
বিঃদ্র:- প্রত্যেক পরীক্ষার জন্য পৃথক বিল করতে হবে।									
অফিসে ব্যবহারের জন্য									
পরীক্ষক						পরীক্ষা নিয়ন্ত্রক			
প্রস্তুতকারী:			পরীক্ষাকারী:		সুপারিশকারী:		অনুমোদনকারী:		

Fig-4.7: After Generating Bill

	A	B	C	D	E
1	Teachers' Name			Teachers' Email	
2	Dr. Md. Shahjahan				
3	Dr. Sk. Md. Masudul Ahsan				
4	Dr. M. M. A. Hashem				
5	Dr. K. M. Azharul Hasan				
6	Dr. Kazi Md. Rokibul Alam				
7	Dr. Muhammad Sheikh Sadi				
8	Dr. Pintu Chandra Shill				
9	Mr. Md. Abdus Salim Mollah				
10	Dr. Sk. Imran Hossain				
11	Mr. Abdul Aziz				
12	Mr. Sunanda Das				
13	Mrs. Nazia Jahan Khan Chowdhury				
14	Mrs. Dola Das				
15	Mr. S. M. Taslim Uddin Raju				
16	Mr. Md. Motaleb Hossen Manik				
17	Mr. Md. Ahsan Habib				
18	Mr. Md. Shahidul Salim				
19	Ms. Dipannita Biswas				
20	Mr. Md. Repon Islam				
21	Mr. Md. Sakawat Hossain				
22	Mr. Md. Nazirulhasan Shawon				
23	Md. Badiuzzaman Shuvo				
24	Most. Kaniz Fatema Isha				
25	Mr. Farhan Sadaf				
26	Mr. Safin Ahmmed				
27	Mr. Argha Chandra Dhar				
28	Mr. Md Mehrab Hossain Opi				
29	Dr. Subrata Talapatra				
30	Mr. Ridwan Mustofa				
31	Mr. Jahid Hasan Ashik				
32	Dr. A. R. M. Jalal Uddin Jamali				
33	Dr. Md. Alhaz Uddin				
34	Dr. S. M. Rabiul Alam				
35	Dr. Md. Hasanuzzaman				
36	Mr. Md. Hasibul Haque				

Fig-4.8: Empty Check Email Sheet

	A	B	C	D	E
1	Teachers' Name			Teachers' Email	
2	Dr. Md. Shahjahan			<a href="mailto:jahan@eee.kuet.ac.bd">jahan@eee.kuet.ac.bd</a>	
3	Dr. Sk. Md. Masudul Ahsan			<a href="mailto:smahsan@cse.kuet.ac.bd">smahsan@cse.kuet.ac.bd</a>	
4	Dr. M. M. A. Hashem			<a href="mailto:hashem@cse.kuet.ac.bd">hashem@cse.kuet.ac.bd</a>	
5	Dr. K. M. Azharul Hasan			<a href="mailto:az@cse.kuet.ac.bd">az@cse.kuet.ac.bd</a>	
6	Dr. Kazi Md. Rokibul Alam			<a href="mailto:rokib@cse.kuet.ac.bd">rokib@cse.kuet.ac.bd</a>	
7	Dr. Muhammad Sheikh Sadi			<a href="mailto:sadi@cse.kuet.ac.bd">sadi@cse.kuet.ac.bd</a>	
8	Dr. Pintu Chandra Shill			<a href="mailto:pintu@cse.kuet.ac.bd">pintu@cse.kuet.ac.bd</a>	
9	Mr. Md. Abdus Salim Mollah			<a href="mailto:salim9326@cse.kuet.ac.bd">salim9326@cse.kuet.ac.bd</a>	
10	Dr. Sk. Imran Hossain			<a href="mailto:imran@cse.kuet.ac.bd">imran@cse.kuet.ac.bd</a>	

Fig-4.9: After Filling Email Address



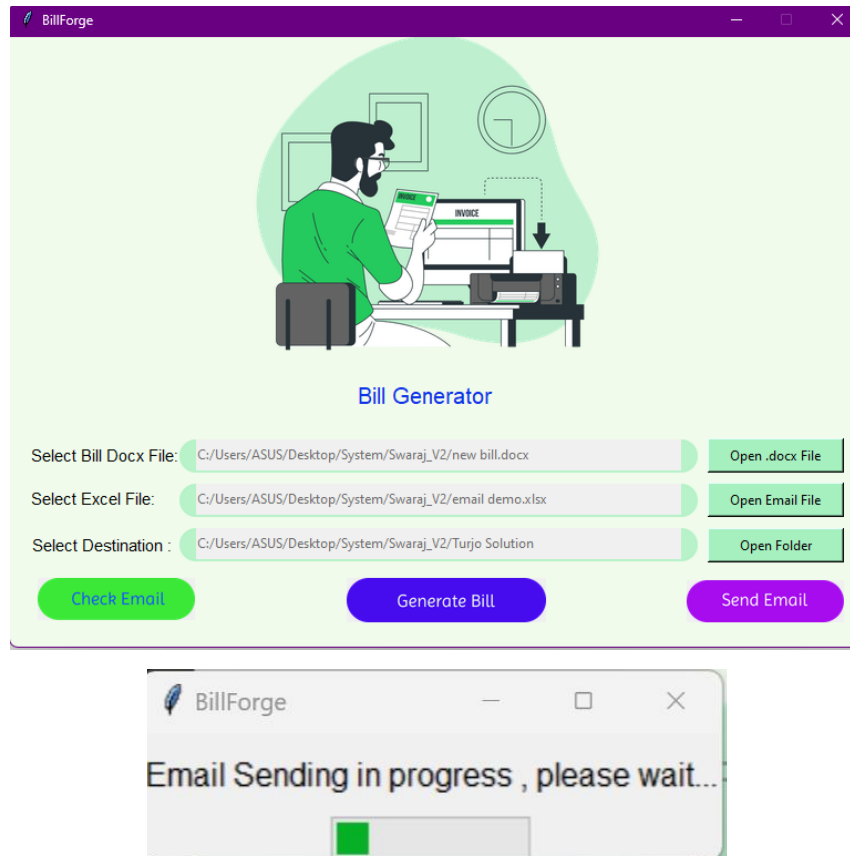


Fig-4.10: Email Sending in Progress

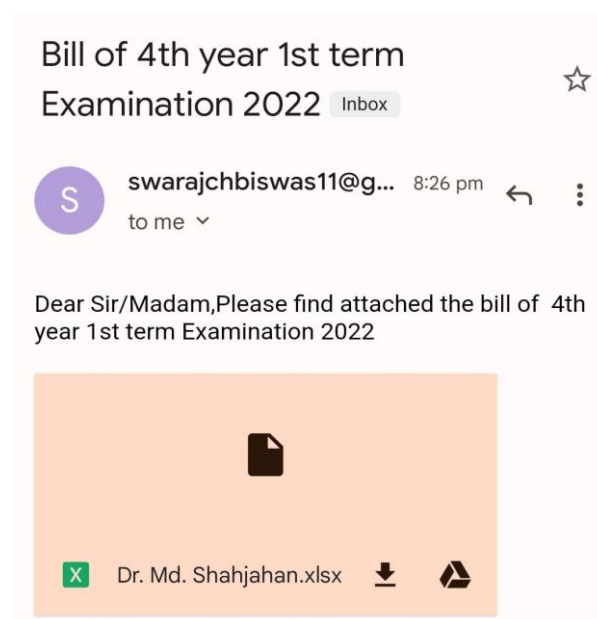


Fig-4.11: Email Successfully Received

List of Bills - 4th year 1st term Examination-2022		
Teacher's Name		Teacher's Bill
Dr. Md. Shahjahan		13500.00
Dr. Sk. Md. Masudul Ahsan		51968.09
Dr. M. M. A. Hashem		34384.00
Dr. K. M. Azharul Hasan		14695.00
Dr. Kazi Md. Rokibul Alam		75158.09
Dr. Muhammad Sheikh Sadi		7024.00
Dr. Pintu Chandra Shill		18029.00
Mr. Md. Abdus Salim Mollah		17804.00
Dr. Sk. Imran Hossain		13624.00
Mr. Abdul Aziz		47618.09
Mr. Sunanda Das		7249.00
Mrs. Nazia Jahan Khan Chowdhury		22238.09
Mrs. Dola Das		54029.09
Mr. S. M. Taslim Uddin Raju		16378.09
Mr. Md. Motaleb Hossen Manik		62393.09
Mr. Md. Ahsan Habib		81083.09
Mr. Md. Shahidul Salim		22583.09
Ms. Dipannita Biswas		33953.09
Mr. Md. Repon Islam		7024.00
Mr. Md. Sakhawat Hossain		7024.00
Mr. Md. Nazirulhasan Shawon		7024.00
Md. Badiuzzaman Shuvo		34328.09
Most. Kaniz Fatema Isha		13133.09
Mr. Farhan Sadaf		7024.00
Mr. Safin Ahmmed		7024.00
Mr. Argha Chandra Dhar		7024.00
Mr. Md Mehrab Hossain Opi		7024.00
Dr. Subrata Talapatra		4500.00
Mr. Ridwan Mustofa		19350.00
Mr. Jahid Hasan Ashik		19350.00
Dr. A. R. M. Jalal Uddin Jamali		2700.00
Dr. Md. Alhaz Uddin		2700.00
Dr. S. M. Rabiul Alam		2700.00
Dr. Md. Hasanuzzaman		2700.00
Mr. Md. Hasibul Haque		2700.00
Total Bill For Teachers:		747040.08

Fig-4.12: Bill Summary

## 4.2 Deployment

This “BillForge” application is not deployed online or any server or anything rather it is locally hosted and offline application which is in focus to the prime users of this system. However, the custom build API is hosted on pythonanywhere.com Anyone who has the login id and password can access this system from any office computers.

### **4.3 Ethical & Moral Issues**

The 'BillForge' application adheres to stringent ethical standards, particularly in prioritizing user privacy by ensuring the confidentiality of user data. The demo billing chart sheet provided with the system is classified and non-disclosable. Therefore, individuals without authorized access are unable to locate or access the file. As there is no exe file the program to security issue remains. Any unauthorized access to the computer can be a threat to that program. Moreover, the teachers receiving emails through this system will only receive data pertaining to themselves, safeguarding their privacy and upholding the commitment to ethical standards.

### **4.4 Financial Analyses and Budget**

This project is designed to operate locally on individual computers, necessitating a compatible system for execution. Given that the system interfaces with Word and Excel documents, and considering that Microsoft Office applications are the standard at Khulna University of Engineering & Technology (KUET), the procurement of enterprise licenses for these products is required. This is a factor in the overall system cost. Transitioning the project to an online platform would incur additional expenses, potentially elevating costs beyond initial estimates. Such a move would require web hosting services and the acquisition of a domain name, contributing further to the financial outlay. Additionally, to maintain system efficiency, a high-speed internet connection is advisable. The investment in the necessary internet infrastructure would thus be a significant component of the project's total cost. Estimated Cost Figure

- System Design Analyst – 40,000 Tk
- System Programmer – 50,000 Tk
- Database Management and Server Handling- 30,000 Tk
- Security Specialist- 30,000 Tk

## **5 Conclusion**

### **5.1 Conclusion & Challenges Faced**

During the development of this system, we encountered several challenges, such as recognizing name patterns and extracting information from DOCX files. Initially, our attempts to extract names and information faced difficulties, particularly when dealing with tables containing various data. The inconsistent naming conventions of teachers in the demo DOCX file further complicated matters, leading to inaccuracies in the algorithm or program results.

To address these issues, we implemented a solution by standardizing the naming conventions. This adjustment significantly improved the accuracy of the algorithm. Subsequently, our focus shifted to the task of reading and writing values from and to Excel sheets, presenting its own set of challenges. One notable obstacle was encountered when attempting to write values to specific cells. To overcome this challenge, we opted to split the sheets, enhancing the efficiency of the process. Additionally, when attempting to send emails from a designated email address using Python code, we faced constraints due to default permission limitations. To resolve this, we generated an application key from Google, granting the necessary permissions and enabling the successful sending of emails through our Python code.

The primary objective of this project is to enhance the efficiency and workflow of office executives across various departments at KUET. This is achieved by automating the process of creating bills for individual teachers in any given semester and subsequently dispatching these bills to the respective teachers. The implementation of this system contributes significantly to the automation of creating and disseminating email notifications regarding teacher bills, streamlining administrative tasks and improving overall efficiency.

## 5.2 Future Study

The automation process of billing system in KUET can be 100% automated when every department and teachers of respective departments are connected together. The use of machine learning technology can be applied to this field to sense the information of different attributes of the teachers and info regarding billing information. Machine learning or artificial intelligence can be applied for recognizing teachers' name, their information and smart algorithm to manage the excel file creation and sending relevant information to the relevant teachers or any different body regarding as per need. The use of personalized database will be added to the system to create a separated user such as teachers who can logged into the system and thus generate the total bill summary or any past history of bill summary. The system can be deployed on the servers to be more dynamic and robust. We will add more features to ensure security such as database encryption and executable file format, flexible document structure and more user-friendly UI and UX.

## 6 References

- [1] AI Sweigart, "Automate Boring Stuff with Python: Practical Programming for Total Beginners 1<sup>st</sup> Edition"
- [2] Python Automation Projects Scripts You Should Know-  
<https://www.freecodecamp.org/news/python-automation-scripts/>
- [3] Automating with Python: A Developer's Guide For 2023-  
<https://www.learnenough.com/blog/automating-with-python>
- [4] Python-docx documentation -<https://www.learnenough.com/blog/automating-with-python>
- [5] OpenPyXL documentation -  
[https://openpyxl.readthedocs.io/en/stable/editing\\_worksheets.html](https://openpyxl.readthedocs.io/en/stable/editing_worksheets.html)
- [6] Tkinter Library Documentation -<https://docs.python.org/3/library/tk.html>
- [7] SMTP protocol documentation- <https://docs.python.org/3/library/smtplib.html>

- [8] Python Documentation -<https://docs.python.org/3/tutorial/classes.html>
- [9] Figma to Tkinter Designer API- <https://github.com/ParthJadhav/Tkinter-Designer>
- [10] For Designing GUI – [www.figma.com](http://www.figma.com)
- [11] For Various Figures- Internet and Novic API and FlatIcon